# THESE DE DOCTORAT DE

L'ECOLE NATIONALE SUPERIEURE
DE TECHNIQUES AVANCEES BRETAGNE
COMUE UNIVERSITE BRETAGNE LOIRE

ECOLE DOCTORALE N° 601
*Mathématiques et Sciences et Technologies de l'Information et de la Communication*
Spécialité : *Télécommunications, Informatique*

Par
**Marwa IBRAHIM**

## Toward Efficient Data Collection and Decision-Making Strategies for Resource-Constrained Sensor Networks

**Thèse présentée et soutenue à Brest, le 14 Décembre 2021**

**Unité de recherche : Lab-STICC UMR CNRS 6285**

### Rapporteurs:

| | |
|---|---|
| Abdelhafid ABOUAISSA | Professeur à l'Université de Haute-Alsace |
| Raja CHIKY | Professeur à l'Institut Supérieur d'Électronique de Paris |

### Composition du Jury :

| | | |
|---|---|---|
| Président : | Christian JUTTEN | Professeur à l'Université Grenoble Alpés |
| Examinateur : | Denis HAMAD | Professeur à l'Université du Littoral Côte d'Opale |
| Dir. de thèse : | Ali MANSOUR | Professeur à l'ENSTA Bretagne |
| Co-dir. de thèse : | Abbass NASSER | Doyen de la Faculté des Sciences et d'Arts à l'AUCE |

### Invités :

| | |
|---|---|
| Hassan HARB | Enseignant-Chercheur à l'AUCE (Co-Encadrant) |
| Christophe OSSWALD | Enseignant-Chercheur à l'ENSTA Bretagne (Encadrant) |
| Isabelle QUIDU | Maitres des conférences (HDR) à l'ENSTA Bretagne (Membre de CSI) |
| Kofi YAO | Maitres des conférences (HDR) à l'UBO (Membre de CSI) |

# ABSTRACT

In today's time, our world faces many risks and dangerous coming from nature, health, military, etc. Thus, there is an essential need to sense everything around us in order to better understand such risks then reduce their effects. Hence, sensing-based technology, particularly the wireless sensor networks (WSNs) and the Internet of things (IoT), have taken a great attention from communities and industries as an efficient and low-cost solution for monitoring various kind of applications. While the potential benefits of such technology is real and significant, two major challenges remain in front of fully realizing this potential: resource-constrained sensors, especially the battery power, and decision making in real-time applications. Subsequently, such challenges are highly related to the huge amount of data collected and transmitted by the sensor nodes, which are mostly redundant, leading, from one hand, to quick consume of their limited battery power and, from other hand, to complicate the mission of experts when analyzing the data. Therefore, designing efficient data collection and decision-making strategies to reduce the size of the raw data collected in such networks is becoming essential to increase their lifetime. In this thesis, we are interested in the cluster-based network topology with the periodic data collection model for reliability and scalability purposes. In such network, each sensor monitors the target area for a certain time period then it sends the collected data to its Cluster-Head (CH) which, in turn, forwards them toward the sink node. Then, we propose several data collection and analysis mechanisms that allow overcoming the limited sensor resources and the big data collection challenges imposed by sensing-based networks. Mainly, the proposed mechanisms work on three network levels (e.g. sensor, CH and sink), and they aim to reduce the amount of data routed in the network while preserving the information integrity at the sink. At the sensor level, we propose data prediction, aggregation and compression methods based respectively on Newton forward difference, divide-and-conquer and elimination similarity algorithms with the aim to reduce the raw data collected by each sensor. At the CH level, we propose new data clustering, fusion, in-network aggregation and scheduling techniques that aim to search the correlation among neighboring nodes then to eliminate the existing data redundancies before sending the data toward the sink. At the sink level, we introduce efficient decision-making models based on customizable user-defined tables that allow end users to analyze the data and make an early decision. We analyzed the performance of our mechanisms based on a set of simulation and experimentations on real sensor data. The obtained results have shown the efficiency of our mechanisms according to energy consumption, data accuracy, and coverage area while improving the performance of sensing-based networks.

**KEYWORDS:** Sensing-based Networks, Energy-Efficiency, Decision-Making, Data Reduction, Spatio-Temporal Correlation, Scheduling Strategies.

# CONTENTS

**Abstract** 1

**Table of Contents** 7

**List of Figures** 10

**List of Tables** 11

**List of Abbreviations** 13

**Notations** 15

**Dedication** 17

**Acknowledgements** 19

**Introduction** 21

1. General Introduction . . . . . 21
2. Main Contributions of this Thesis . . . . . 22
3. Thesis Structure . . . . . 24

**1 An Overview About Sensing-Based Networks** 27

1.1 Introduction . . . . . 27
1.2 Sensing-based Network Architecture . . . . . 28
1.2.1 Sensor Node Architecture . . . . . 28
1.2.2 Sensor Network Architecture . . . . . 29
1.3 Types of Sensing-Based Networks . . . . . 29
1.4 Survey on Sensing-Based Applications . . . . . 32
1.4.1 Military Surveillance . . . . . 32
1.4.2 Environment Monitoring . . . . . 33
1.4.3 Healthcare Monitoring . . . . . 33
1.4.4 Industrial Monitoring . . . . . 33
1.4.5 Agricultural Monitoring . . . . . 34

1.4.6 Robots Monitoring . . . 34
1.4.7 Water and Ocean Monitoring . . . 35
1.5 Network Design . . . 35
1.5.1 Cluster-Based Architecture . . . 35
1.5.2 Periodic Data Collection Model . . . 36
1.6 Sensor Network Challenges . . . 37
1.6.1 Deployment . . . 37
1.6.2 Energy Consumption . . . 38
1.6.3 Security . . . 38
1.6.4 Scalability . . . 38
1.6.5 Routing . . . 39
1.6.6 Coverage . . . 39
1.6.7 Big Data Collection and Management . . . 40
1.6.7.1 At Sensor Node . . . 40
1.6.7.2 At CH Node . . . 40
1.6.7.3 At Sink . . . 41
1.7 Energy-Efficiency and Data Reduction: A Background . . . 41
1.7.1 Aggregation-based Techniques . . . 41
1.7.2 Compression-based Techniques . . . 42
1.7.3 Prediction-based Techniques . . . 43
1.7.4 Clustering-based Techniques . . . 44
1.7.5 In-Network based Techniques . . . 46
1.7.6 Adapting-based Sensing Frequency Techniques . . . 47
1.7.7 Scheduling-based Techniques . . . 48
1.7.8 Decision-Making based Techniques . . . 49
1.8 Conclusion . . . 50

**2 ON-IN: An On-Node and In-Node Based Mechanism for Big Data Collection in Large-Scale Sensor Networks 51**

2.1 Introduction . . . 51
2.2 Sensor Level: On-Node Prediction Model . . . 51
2.2.1 Newton's Forward Difference Method . . . 52
2.2.2 On-Node Prediction Algorithm . . . 52
2.3 CH Level: In-Node Clustering Model . . . 53
2.3.1 Pattern-K-means Algorithm: PK-means . . . 53

2.4 Performance Evaluation . . . 55
2.4.1 Simulation Results . . . 55
2.4.1.1 Raw Data vs Recovered Data . . . 56
2.4.1.2 Sensor Data Transmission Ratio . . . 56
2.4.1.3 CH Data Transmission Ratio . . . 57
2.4.2 Experiment Results . . . 58
2.4.2.1 Raw Data vs Recovered Data . . . 58
2.4.2.2 Iteration Loop Number . . . 59
2.5 Conclusion . . . 60

**3 Adaptive Strategy and Decision-Making Model for Sensing-Based Network Applications 61**
3.1 Introduction . . . 61
3.2 Sensor Tier: Divide-and-Conquer Algorithm . . . 62
3.2.1 Divide-and-Conquer Algorithm . . . 62
3.2.2 Illustrative Example . . . 62
3.3 CH Tier: Support-Confidence Method . . . 63
3.3.1 Support-Confidence Algorithm . . . 63
3.3.2 Illustrative Example . . . 64
3.4 Sink Tier: Decision-Making Model . . . 65
3.4.1 Real-Time Decision Model . . . 65
3.5 Simulation Results . . . 66
3.5.1 Data Reduction Ratio at Sensors . . . 66
3.5.2 Data Reduction Ratio at CHs . . . 68
3.5.3 Decision Results at the Sink . . . 69
3.6 Conclusion . . . 70

**4 All-in-One: Toward Hybrid Data Collection and Energy Saving Mechanism in Sensing-Based Applications 71**
4.1 Introduction . . . 71
4.2 An Overview to All-in-One Mechanism . . . 72
4.3 On-Period Redundancy Elimination Model . . . 72
4.3.1 Aggregation-Based Reduction Technique . . . 73
4.3.2 Compression-Based Reduction Technique . . . 74
4.3.3 Prediction-Based Reduction Technique . . . 75
4.3.4 Performance Discussion of On-Period Techniques . . . 76

4.3.4.1 Selection of Thresholds' Values . . . 76
4.3.4.2 Accuracy Study . . . 76
4.3.4.3 Complexity Study . . . 76
4.3.4.4 Energy Consumption Study . . . 77
4.3.5 Hybrid-Based On-Period Reduction Technique . . . 77
4.3.5.1 ANOVA Model and Bartlett Test . . . 78
4.3.5.2 Sensor Battery Level . . . 78
4.3.5.3 On-Period Data Decision . . . 79
4.4 In-Period Redundancy Elimination Model . . . 79
4.4.1 Sensing Frequency Adaptation (SFA) Mechanism . . . 80
4.4.2 On-Off Transmission (OOT) Mechanism . . . 80
4.4.3 Hybrid-Based In-Period Reduction Technique . . . 81
4.4.3.1 In-Period Similarity Study . . . 82
4.4.3.2 In-Period Decision Table . . . 82
4.5 In-Node Redundancy Elimination Model . . . 83
4.5.1 In-Network Aggregation Approach . . . 83
4.5.2 Data Clustering Approach . . . 84
4.5.3 Hybrid-Based In-Node Reduction Technique . . . 85
4.6 Simulation Results . . . 86
4.6.1 On-Period Decision Study . . . 86
4.6.2 In-Period Decision Study . . . 88
4.6.3 Data Transmission Ratio at Sensor . . . 88
4.6.4 Energy Consumption in Sensor . . . 89
4.6.5 Packet Types Study at CH . . . 90
4.6.6 In-Node Decision Study . . . 91
4.7 Conclusion . . . 94

**5 Aggregation-Scheduling Based Mechanism for Energy-Efficient Multivariate Sensor Networks 95**

5.1 Introduction . . . 95
5.2 AGING Mechanism . . . 96
5.2.1 Aggregation Phase . . . 96
5.2.1.1 Score-based Table . . . 97
5.2.1.2 Multi-Aggregation Technique . . . 97
5.2.2 Scheduling Phase . . . 98

5.2.2.1 Graph-based Spatio-Temporal Node Correlation . . . . . 98
5.2.2.2 Graph Construction and Nodes-Coloring . . . . . 99
5.2.2.3 Disjoint Sets and Node Scheduling . . . . . 100
5.3 Performance Evaluation . . . . . 100
5.3.1 Data Transmission Ratio at Node Level . . . . . 101
5.3.2 Average Node Lifetime . . . . . 101
5.3.3 Percentage of Data Loss . . . . . 102
5.3.4 Active Nodes vs Zone Coverage . . . . . 103
5.4 Conclusion . . . . . 105

**6 Conclusions and Perspectives 107**
6.1 Conclusions . . . . . 107
6.2 Perspectives . . . . . 108
6.2.1 Short to Mid Term Perspectives . . . . . 108
6.2.2 Long Term Perspectives . . . . . 109

**Publications 111**

**Bibliography 125**

# List of Figures


1.1 Sensor node architecture. . . . 29
1.2 Typical sensing-based networks: single-hop vs and multi-hop communication. . . . 30
1.3 Types of sensing-based applications. . . . 31
1.4 Two-layers cluster-based architecture network. . . . 36

2.1 Distribution map of the sensors in the Intel lab. . . . 56
2.2 Comparison between raw and Newton Gregory generated data, $\mathcal{T} = 100$ . . . 57
2.3 Number of readings periodically sent to the CH. . . . 57
2.4 Periodic number of sets sent to the sink, $d = 6$. . . . 58
2.5 Distribution of nodes in our lab. . . . 59
2.6 Comparison between raw and recovered data, $d = 6, \mathcal{T} = 100$. . . . 59
2.7 Number of iterations in applying PK-means, $\mathcal{T} = 100, d = 6, K = 3$. . . . 60

3.1 Illustrative example for divide-and-conquer algorithm. . . . 63
3.2 Illustrative example for support-confidence algorithm. . . . 65
3.3 Customizable score table. . . . 66
3.4 Early decision table. . . . 66
3.5 Distribution of sensor nodes and CHs in the Intel Lab. . . . 67
3.6 Number of periodic readings sent from each sensor to the CH. . . . 67
3.7 Size of periodic data sent from each CH to the sink. . . . 68
3.8 Score table customized to temperature monitoring. . . . 69
3.9 Early decision table customized to temperature monitoring. . . . 69
3.10 Variation of decision-making at the sink during periods, $\mathcal{T} = 100, \mathcal{V} = 5, \mathcal{C} = 10$. . . . 70

4.1 Flow diagram of All-in-One mechanism. . . . 73
4.2 Variation of the on-period technique selected by the sensor at each period, $\mathcal{T} = 50$, $\epsilon = 0.1$, $\rho = 0.5$, $d = 5$. . . . 87
4.3 Variation of the in-period decision made by the sensor at each round, $\mathcal{T} = 50$, $\epsilon = 0.1$, $\rho = 0.5$, $d = 5$. . . . 88

4.4 Variation of the sensing frequency of a sensor during periods, $\mathcal{T} = 50$, $\epsilon = 0.1$, $\rho = 0.5$, $d = 5$. . . . . 89
4.5 Number of readings sent from each sensor to the CH. . . . . 90
4.6 Remaining energy in a sensor in function of the period progress, $\mathcal{T} = 50$, $\epsilon = 0.1$, $\rho = 0.5$, $d = 5$. . . . . 91
4.7 Variation of periodic packet types received by the CH. . . . . 92
4.8 Number of sets sent periodically from the CH to the sink. . . . . 93
4.9 Illustrative example of packet types received by the CH during a period and after applying K-means over the compressed packets, $K = 4$. . . . . 94

5.1 Architecture of AGING mechanism. . . . . 97
5.2 Illustrative example of node correlation and scheduling. . . . . 100
5.3 Average percentage of data sent from each node to the CH at each period. 102
5.4 Average lifetime of each node. . . . . 103
5.5 Percentage of data loss after applying scheduling strategies. . . . . 104
5.6 Variation of the coverage zone area in function of the number of active nodes, $\mathcal{T} = 500, \delta_j = 0.05, G = S_r, K = 3$. . . . . 105

# List of Tables

2.1 Simulation parameters. . . . . 56

4.1 On-period data decision table. . . . . 79
4.2 In-period data decision table. . . . . 83
4.3 Simulation environment. . . . . 86

5.1 Simulation environment. . . . . 101

# ABBREVIATIONS

**ANOVA** ANalysis Of VAriance

**CH** Cluster-Head

**IoT** Internet of Things

**NFD** Newton's Forward Difference

**OOT** On-Off Transmission

**PCC** Pearson Correlation Coefficient

**PFF** Prefix Frequency Filtering

**PK-means** Pattern K-means

**PPMC** Pearson Product-Moment Coefficient

**PSN** Periodic Sensor Network

**QoS** Quality-of-Service

**SFA** Sensing Frequency Adaptation

**SFDC** Structure Fidelity Data Collection

**S-LEC** Sequential Lossless Entropy Compression

**WBSN** Wireless Body Sensor Network

**WSN** Wireless Sensor Network

# NOTATIONS

| | |
|---|---|
| $\alpha_0$, $\alpha_1$ | ANOVA thresholds |
| $C$ | Confidence threshold in support-confidence algorithm |
| $\delta_j$ | Aggregation threshold in score table |
| $d$ | Number of selected data points in NFD |
| $\mathcal{D}_{i1}^{p}$ | Subset of selected points for node $i$ during period $p$ in NFD |
| $\epsilon$ | $Aggregate$ threshold |
| $E_c$ | Critical energy threshold |
| $E_i$ | Initial sensor energy |
| $F^p$ | Frequent mean set in support-confidence algorithm |
| $G$ | Geographical threshold |
| $h$ | Interval of difference in NFD |
| $K$ | Number of clusters |
| $n$ | Number of sensor node in a network |
| $N$ | Sensor network |
| $N_i$ | Sensor node number $i$ |
| $M_i^p$ | Matrix of readings collected by all sensors of node $i$ during a period $p$ |
| $p$ | Period time |
| $\pi$ | Round size |
| $\mathcal{P}_{i1}^{p}$ | Statistical parameters of a reading set $R_{ij}^p$ when appyling PK-means |
| $q$ | Number of points in NFD |
| $Q$ | Number of sensors in each sensor node |
| $\rho$ | Pearson's threshold |
| $r_l$, $r_u$ | Lower and upper bounds of the normal range in customizable score table |
| $r_t^{ij}$ | Reading collected by the sensor $S_{ij}$ during the slot time $t$ |
| $R_{ij}^p$ | Set of readings collected by the sensor $j$ in node $i$ during the period $p$ |

$R_{i1}^{'p}$ Set of readings sent from a node $N_i$ to the CH at the end of each period $p$

$\sigma_p^2$ Pooled variance

$Sup(m_t)$ Support of the mean value $m_t$

$S_{ij}$ Sensor number $j$ in the node number $i$

$t$ Slot time number $t$in a period time

$t_E$ Eulidean distance threshold

$t_J$ Jaccard threshold

$T$ The variance resulted from ANOVA

$T_\alpha$ Critical value of ANOVA according to a false-rejection probability $\alpha$

$\mathcal{T}$ Total number of readings collected by each sensor during a period $p$

**(**$x_i$**,**$y_i$**)** Coordinates of point number $i$ in NFD

$\mathcal{V}$ Number of divisions

$wgt(r_t)$ Weight of the reading $r_t$

# DEDICATION

*I am dedicating this work to my beloved parents '**IBRAHIM & FERYAL**' in hope that my achievements will make you proud and pay you back a little bit for all what you did to us and all what you went through. For my family, my friends and my teachers, for the most amazing doctors who stood by my side and influenced me positively throughout those years, I can't thank you enough for your unconditional love and support and for always believing in me even when I couldn't believe in myself. You made this journey worth going! This work is the fruit of your presence and therefore, I dedicate it to you. I will always be grateful for your presence.*

# ACKNOWLEDGEMENTS

First and foremost, I would like to thank **Allah** for giving me the power to undergo this work until its completion and for blessing me with all his generous blessings. Then, I thank all the people who followed and contributed with me to produce this work.

I would like to express my sincere appreciation and gratitude to my supervisors: Prof. Ali MANSOUR and Dr. Christophe OSSWALD, at the ENSTA-Bretagne, and Dr. Abbass NASSER and Dr. Hassan HARB, at the American University of Culture and Education (AUCE), for their guidance during my research. Their support, efforts and inspiring suggestions were essential to the birth of this document and to my formation as a future researcher. They have been a constant source of encouragement and enthusiasm, and I appreciate all their contributions of times, ideas, and funding to make my Ph.D productive and stimulating.

Moreover, I would like to thank the president and the members of the jury committee: Prof. Christian Jutten, Prof. Abdelhafid Abouaissa, Prof. Raja Chiky, Prof. Denis Hamad, Dr. Isabelle Quidu, and Dr. Kofi Yao for their time, interest, and helpful comments. It is an honor to have my work examined and assessed by professional experts such as yourselves.

Additionally, I am very gratefully to all people I have met along the way and have contributed to the development of my research. My appreciation and thanks go to all members of the Lab-STICC for their moral and technical support at all times. A special thank goes to Mdm. Annick BILLON-COAT for all the received assistance during my study.

Lastly, my deepest gratitude goes to my family for their unflagging love and unconditional support throughout my life and my studies. For my parents, Ibrahim and Feryal, who raised me with a love of science and supported me in all my pursuits. You made me live the most unique, magic and carefree childhood that has made me who I am now. For my brothers Ali and Abbass, and my sisters Sahar, Suzan, and Mariam whose faithful support during this Ph.D. The support and the encouraging of all members of my big family are so appreciated.

# INTRODUCTION

## 1. GENERAL INTRODUCTION

Nowadays, we are living in a world full of risks where each of them has its own form, severity and impact. Particularly, the nature, the health and the military sectors are the most serious threats we face in today's world. First, the nature is the source of many disasters such as flood, volcanic eruptions, earthquakes, etc. which are increasing day after day due the climate change. Second, the wars historically produced a severe loss in people and economic in which, possibly, the use of biological, chemical and nuclear weapons leads to a dramatic end to humanity. Finally, the rapid emergence of viruses and illnesses constitutes a major challenge for governments and industries where their propagation leads, mostly, to thousands of victims before an appropriate treatment is found. Therefore, the governments started, from the beginning of the third millennium, to collect everything in our planet in order to understand the nature and the people behaviors thus try to reduce the risk impacts.

Recently, sensing-based networks have taken a great attention from both industries and researchers thanks to the great developments in communication technologies. They offer effective and low cost systems that allow remotely monitoring the target areas and periodically sending the collected data to the sink node for a later analysis and study purposes. Typically, a sensing-based network consists of a large set of sensor nodes that sense physical phenomena, process the raw data and transmit them through wireless communication toward the end user. Nowadays, one can find various types of sensor devices and, from their appearance until the present day, WSN have had an exponential increasing range of applications such as military, agriculture, industrial, home automation, healthcare, weather, underwater, etc. [1].

Indeed, researchers have to face many challenges related to the design of sensing-based networks. Some fundamental challenges include the sensor deployment, the overall coverage of the monitored zone, the network scalability, the short communication range, the node failure, etc. However, energy conservation emerges is considered as one of the most critical design issues in hardware and software for such networks. On one hand, the network lifetime is highly related to the sensor energy power that is mostly equipped with a small battery, which cannot be replaced or recharged in harsh environments. On the other hand, the energy consumption is heavily dependent on the amount of data collected in the network; the more the data are collected and transmitted, the more the energy is consumed. Therefore, designing new data collection techniques are becoming a fundamental operation in sensor networks in order to conserve the network energy . The idea behind such techniques is to search the similarities among the collected data in the network then try to reduce the amount of data collection and transmission along the path to the sink.

In this thesis, we propose energy-efficient data collection mechanisms dedicated to

sensing-based applications based on clustering architecture. More specifically, we focus on data reduction mechanisms at all network sides (e.g. sensor, CH and sink) with the main goal of extending the network lifetime and making a real time decision. Subsequently, we propose several techniques that manage data collected/transmitted in each cluster, where appropriate algorithms have been applied at every network side. Our proposed techniques are validated via both simulations and experimentations on real sensor data and comparison with other existing data reduction techniques. The results show that the effectiveness of our techniques in terms of improving the performance of the network, extending its lifetime, and making a fast decision while taking into account the requirements of the monitored application.

## 2. Main Contributions of this Thesis

The main contributions in this thesis concentrate on designing energy-efficient data collection techniques at sensor and CH nodes, and a decision making models at the sink under a cluster-based sensing applications.

**A) At Sensor Node:** In sensing-based applications, the sensor node constitutes the main component in the network that continuously monitors the zone target and periodically sends the collected data to the sink. Subsequently, such periodic collection along with the dense sensor deployment required in most applications lead to a huge redundancy among the collected data. Such sensor redundancy is generated either by data collected during each period, called as in-period redundancy, or among successive periods, called as on-period redundancy. Thus, it becomes necessary to search, then eliminate, the data redundancy at the sensor level in order to reduce its transmission and thus, enhancing its energy consumption. In this thesis, we propose several techniques that allow reducing the data redundancy in both on-period and in-period. The proposed techniques are described as follows:

1. **On-Period Techniques:** In this step, we are interested in eliminating data redundancy through three main techniques: data aggregation, data prediction and data compression. First, data aggregation has an objective to study the similarities among the collected data, eliminate the existing redundancy and deliver a useful information to the end user in order to take a suitable decision. In this thesis, we propose three data aggregation techniques based respectively on divide-and-conquer algorithm, threshold-based method, and score-based multi-aggregation table. Second, the data prediction allows each sensor to build, based on the periodic collected data, a predictive model in order to send to the sink which, in its turn, regenerates the raw data. This thesis proposes a new data prediction technique based on the Newton's forward difference method that limits the sensor transmission to a set of coefficient values instead of sending the whole raw data. Third, data compression is the process of combining redundant readings into a reduced set of records thus it reduces the packet size periodically sent to the sink. In this thesis, we propose a data compression technique that allows searching the correlation, based on the Pearson coefficient, among readings collected at each period then try to compress them before sending to the CH.

2. **In-Period Techniques:** Obviously, the redundancy level among the collected data is highly dependent on the variation of the monitored condition. For instance, the monitoring of weather temperature or humidity will produce a high redundancy level because such conditions are slowly varying during the progress of periods. Thus, in-period data redundancy should be also eliminated in order to further conserve the sensor energy. Mainly, the in-period data redundancy can be eliminated thanks to two main techniques: sensing frequency adaptation and on-off transmission. From one hand, a common way to perform data reduction is by deploying sensing adaptation mechanism at the sources, e.g. the sensors, while regenerating the reduced amount of data at the sink. This can minimize the data routed in the network and save the sensor battery power. In this thesis, we introduce an adaptive sensing model that studies the data collected by a sensor during a round of periods then adapts its sensing frequency according to several criteria such as the condition variation, its remaining battery level, the correlation with other nodes, etc. On the other hand, on-off strategy aims to switch off the sensor transmission in case a high similarity among data collected in successive periods is detected. This can help in saving the sensor battery and decreasing the packet congestion in the network. This thesis proposes an on-off transmission mechanism, based on the similarity function, that allows to avoid sending similar data in successive periods from each sensor to the CH.

**B) At CH Node:** In sensing-based networks, the sensors are mostly scattered in a random way through aircraft or rocket due to the harsh or inaccessibility of most monitored zones. This leads to a high level of spatial-temporal correlations between sensor nodes. Thus, when receiving the data sets from all sensors at the end of each period, the CH can benefit from such correlations in order to eliminate the data redundancy among neighboring sensors, e.g. in-node data redundancy, before sending them toward the sink. Therefore, the periodic data transmitted by the CH will be reduced which will save its energy and facilitate the data analysis task of the end user. In this thesis, we are interested in four data reduction approaches to remove in-node data redundancy at the CH level: data clustering, in-network data aggregation, data fusion, and scheduling. First, data clustering is the process of grouping nods generating periodic redundant data into clusters then selecting a set of sensor data to send to the sink instead of the whole sensor data. In this thesis, we propose two data clustering techniques, e.g. PKmeans and Hybrid-based clustering, that aim to reduce the redundancy among data generated by neighboring nodes. Second, in-network data aggregation aims to search the similarities among every pair of sensor datasets then to select a subset of datasets to send to the sink while eliminating the other ones. This thesis propose an in-network data aggregation based on the Euclidean distance and consists of two steps, e.g. pairs generation and pairs selection, to reduce the in-node data redundancy at the CH. Third, data fusion has an objective to combine data collected by several sensors into one useful information before sending to the sink. This thesis introduces a new data fusion technique based on the support-confidence method that allows CH to combine data generated by the sensors in its cluster into non-redundant information representing the status of the monitored condition at that cluster. Finally, scheduling is the process of switching sensors in the network into sleep/active modes according to the correlation between them. This will result in saving the energy in both

sensors and CH as well as to reduce the network congestion and minimize the huge data size transmitted to the sink. In this thesis, we propose a new scheduling strategy that allows CH to reduce the sensor activities according to two steps. The first step is to search the spatial-temporal correlation between sensors in the same cluster. Then, in the second step, we propose a strategy based on map coloring and disjoint sets to select a subset of sensors to switch them into active mode while switching-off the others into sleep mode.

**C) At Sink Node:** Decision-making is the main target behind deploying sensing-based applications and it is considered as the last fundamental operation that allows end users to make a decision based on the received data. However, the massive amount of data collected in periodic model, which are mostly redundant, along with the huge number of deployed sensors, which are mostly correlated, required in some applications make the decision-making process a complicated task for the end users. In addition, the decision-making is a critical operation that can, sometimes, lead to a crucial consequences, especially in critical applications such as healthcare and military. For instance, a wrong decision by the medical team may lead to the death of the patient or a mistake in the decision taken by farmer about the farm irrigation may lead to a waste of water. Unfortunately, most of researchers have focused on overcoming challenges exposed by sensing-based applications like network lifetime, sensor localization, security, etc. while few of them were interested to propose decision-making techniques at the sink level. In this thesis, we propose a real-time decision-making model that may be customized depending on the context and circumstances of the monitored application. Our model is an expert-defined and it is based on two customizable tables, e.g. score decision table and early decision table, that are used by the application services staff to determine the real-time status of the monitored zone.

## 3. THESIS STRUCTURE

The thesis is structured as follows:

**Chapter 1: An Overview About Sensing-Based Networks:** This chapter presents a general review about sensing-based networks, their architecture as types, as well as the importance of this domain through various types of applications. Then, it introduces the cluster-based topology combined with periodic data collection model as an efficient architecture for such networks. Finally, it shows the challenges that face the implementation of sensing-based networks while highlighting the energy consumption and the big data collection as the major challenges in such networks.

**Chapter 2: ON-IN: An On-Node and In-Node Based Mechanism for Big Data Collection in Large-Scale Sensor Networks:** This chapter introduces a new data collection mechanism called ON-IN that works at both sensor and CH levels, and aims to reduce the amount of data routed on the network thus, improving its lifetime. At the sensor level, the mechanism proposes a data prediction technique based on the Newton's forward difference method to reduce the amount of raw data sent toward the sink. At the CH level, the mechanism introduces a data clustering technique based on a new variant of Kmeans algorithm, called as PKmeans

(Pattern-Kmeans), in order to reduce the redundancy among data generated by neighboring nodes.

**Chapter 3: Adaptive Strategy and Decision Making Model for Sensing-Based Network Applications:** This chapter introduces an efficient data collection and energy conservation mechanism that is dedicated to reduce the energy cost of data transmission in sensor networks. We present a three-level data reduction mechanism based on three techniques: data aggregation, data fusion and decision-making. The data aggregation technique is applied on the sensor node itself and it uses a divide-and-conquer algorithm that aims to send a reduced set of data from the sensor to its appropriate CH. The data fusion technique is applied at the CH and it is based on support-confidence method that combines data coming from neighboring nodes before sending to the sink. Finally, the decision making model is applied at the sink and allows end users to make a real-time decision based on two customizable tables.

**Chapter 4: All-in-One: Toward Hybrid Data Collection and Energy Saving Mechanism in Sensing-Based IoT Applications:** In this chapter, we propose another energy-efficient data collection mechanism called All-in-One that takes advantages from existing data reduction techniques to more improve the network lifetime. The proposed mechanism works on three main phases (e.g. on-period, in-period and in-node) with the aim to make a trade-off between data reduction techniques on each phase. During on-period phase, we introduce and compare three data reduction methods (e.g. data aggregation, compression and prediction) that allow reducing the data collected by each sensor during a period. During in-period phase, we make a trade-off between two data reduction methods (e.g. on-off transmission and adapting sensing frequency) that allow reducing the amount of data sent by each sensor in successive periods. Finally, the in-node phase is applied at the CH and aims to remove the redundancy among data collected by neighboring nodes, based on in-network correlation or data clustering techniques, before sending the data to the sink.

**Chapter 5: Aggregation-Scheduling Based Mechanism for Energy-Efficient Multivariate Sensor Networks:** This chapter is dedicated to explore data correlation between neighboring nodes in multivariate sensor networks. We propose a two-tier scheduling-based strategy that allows reducing the power activity of the nodes in the network without losing the information integrity or the zone coverage. At the node tier, our mechanism introduces a multi-aggregation technique that allows reducing the node-CH communication based on a customizable score table. At the CH tier, we propose a scheduling strategy that allows searching the spatio-temporal correlation between nodes then to switch some of them into sleep/active mode based on graph coloring and disjoint sets algorithms.

**Chapter 6: Conclusion and Perspectives:** This chapter concludes our work and highlights some aspects of suggested future research work.

# 1

# An Overview About Sensing-Based Networks

## 1.1/ Introduction

Nowadays, the number of connected sensor devices are widely increased and largely exceeding the population number. In everyday life, one can find a huge number of deployed sensors in various applications collecting many kinds of data. Indeed, surveillance, data collection and sensing have been recently introduced in various applications, such as: military, agriculture, environments, industrial, home automation, transport, etc. [2, 3, 4]. Whilst data collected by such sensors can take values, images, audio or video types depending on the application requirements. Starting from the beginning of this decade, sensor devices have been more and more organized into networks under different communication protocols referred as sensing-based networks. Mainly, Internet of Things (IoT) and Wireless Sensor Networks (WSN) constitute the pillars of such networks. With sensing-based networks, we are able to monitor anything at anywhere and anytime for analyzing and studying purposes.

Indeed, sensing-based applications are facing several challenges and problems caused by the limited sensor resources and the densely deployment of the devices. However, major challenges for researchers become how to deal, store and analyze a huge amount of data collected in such networks. Furthermore, the sensor devices are energy-constrained and recharging their batteries is not always an option and it may become a costly operation. In addition, data transmission is the higher energy cost in the sensor that quickly depletes its available power and lowers its lifetime. Hence, data reduction approach has taken a great attention from researchers in order to overcome the big data challenges imposed by sensing-based networks. The main objective of such approach is to minimize the data transmission in the network by removing on-node and in-node redundancy existing among the collected data. In the literature, data reduction takes several forms such as aggregation, compression, prediction, sensing rate adaptation, clustering, etc. However, the selection of a suitable technique is highly related to the targeted application and the desired performance metric (energy consumption, data accuracy, complexity, etc.) that must be optimized.

## 1.2/ SENSING-BASED NETWORK ARCHITECTURE

In this section, we first present the architecture of each sensor node; then we introduce the architecture of the global network.

### 1.2.1/ SENSOR NODE ARCHITECTURE

The sensor node is the basic element of the sensor networks that performs three main operations: sensing, processing and communication. Consequently, each sensor node is composed of four main components and, eventually, of three secondary components that can be added if requested [5] (Figure 1.1). From one hand, the main node units are described as follows:

**(I) Sensing Unit:** it is responsible for sensing physical phenomena and consists of the “sensors” and the “Analog to Digital Converters” (ADCs). The sensors produce analog signal that are converted into digital data by ADC, then sent to computation unit for more processing.

**(II) Computation Unit:** it is responsible for managing all sensing, communication, and self-organization instructions, in order to allow sensor nodes to cooperate together during sensing tasks. This unit is consisting of processer chip, active short-term memory for storing sensed data, an internal flash memory for storing program instructions and an internal timer.

**(III) Communication Unit:** it is responsible for data transmission and reception performed by the transceiver circuitry. This circuitry consists of a mixer, frequency synthesizer, voltage-controlled oscillator (VCO), phase-locked loop (PLL), demodulator, and power amplifiers, all of which consume valuable power.

**(IV) Power Unit:** it is responsible for supplying all units.

On the other hand, the additional component units are described as follows:

**(I) Localization System:** in many sensing applications, the proposed algorithms, such as routing and sensing coverage algorithms, need information about the location of the sensor nodes. The localization system consists of Global Positioning System (GPS) or discovery algorithm that gives information about location using distributed computation.

**(II) Mobilizer:** it is responsible for moving the sensor node from one location to another to perform certain task. This movement is controlled by the mobility function in cooperation with the sensing and computation unit.

**(III) Power Generator:** it is responsible for prolonging the network lifetime in case certain applications need to operate beyond the expected time.

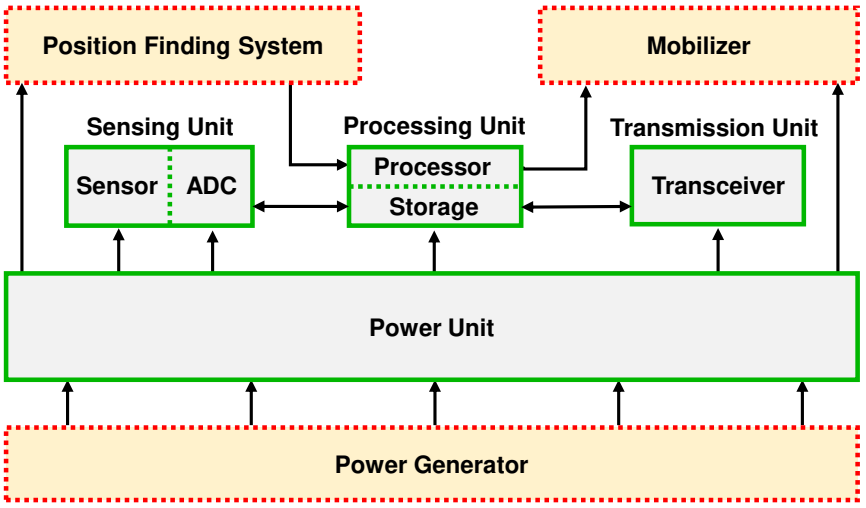



Figure 1.1: Sensor node architecture.

### 1.2.2/ Sensor Network Architecture

Commonly, a sensor network consists of a huge number of sensor nodes deployed in the target area with one or more than a sink node. The sink sends request commands to sensor nodes in a target area; then these nodes should collaborate with each other to accomplish the sensing task and send collected data to the sink or base station. Afterwards, the data are sent to the end-user through the Internet (Figure 1.2). Mostly, the sensor nodes send their data to the sink either through single-hop or mutli-hop communications; in the single-hop communication, the nodes are configured into star topology and they send their data directly to the sink node for long distance communication (Network 2 in Figure 1.2). While, in multi-hop communication, sensor nodes are grouped into a mesh topology, where their data are forwarded from a node to another before reaching the sink node (Network 1 in Figure 1.2). Subsequently, multi-hop communication is more used in sensor applications for energy conservation, efficient transmission and scalability purposes.

## 1.3/ Types of Sensing-Based Networks

The sensing-based technology has many potential applications due to the existence of wide diversity of sensor nodes having various characteristics and types. In the literature, we can distinguish among of sensing-based applications (Figure 1.3):

**(I) Terrestrial-based applications:** in this kind of applications, the network is composed of a high number (sometimes arrived up to hundreds of thousands) of sensor nodes over the monitored land. Unfortunately, such dense deployment leads to the problem of data redundancy and network overload, especially in a random sensor deployment through a rocket or an aircraft. Consequently, proposing new elimination techniques becomes essential in this type of network such as short transmission range, multi-hop optimal routing, in-network data collection, and using low duty-cycle operations. Examples of terrestrial-based applications include the moni-

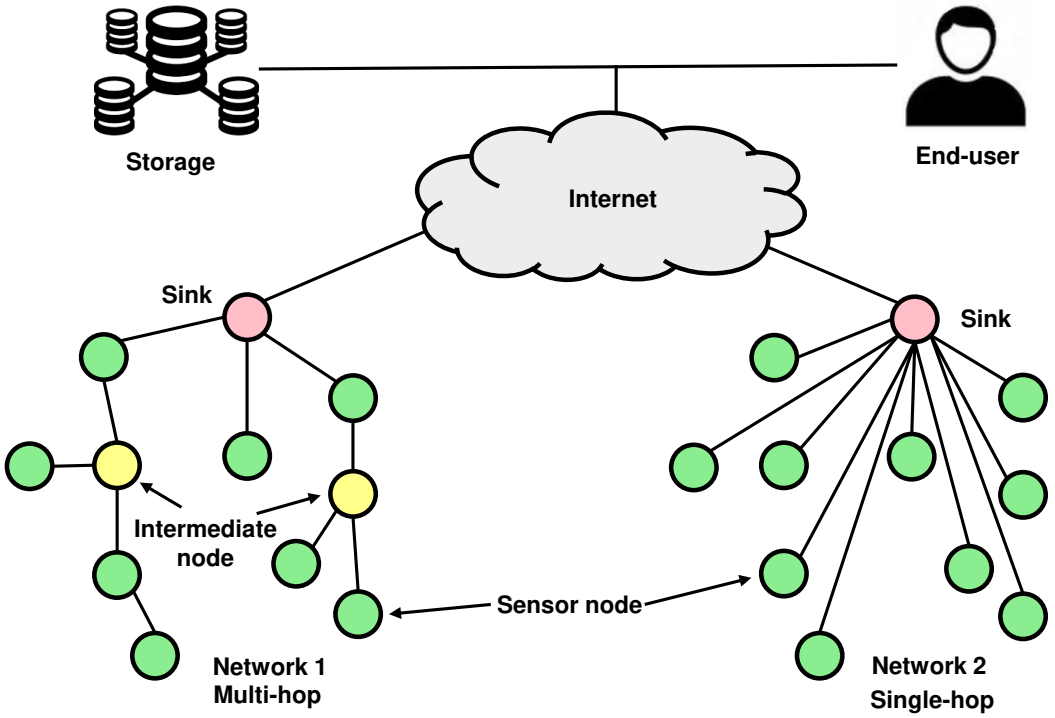



Figure 1.2: Typical sensing-based networks: single-hop vs and multi-hop communication.

toring of environment, industrial, healthcare and security [1, 6, 7].

**(II) Underground-based applications:** this type of networks is composed of a number of sensor nodes that are deployed under the ground or in caves to monitor the underground conditions. Data gathering from the sensor nodes under the ground need additional sensor nodes act as sink nodes located above the ground to send the sensed data to the base station. This type of nodes has higher cost than other sensor like terrestrial networks because the nodes must provide additional functions to guarantee the communication reliability as it passes through soil, rocks, and water. Examples of underground applications are structural monitoring, agriculture monitoring, landscape management, underground environment monitoring of soil, water or mineral and military border monitoring [8, 9].

**(III) Underwater-based applications:** the underwater monitoring has taken a great attention from scientific communities since oceans cover about three fourth of the earth surface. Mainly, the underwater network consists of a set of acoustic sensors and vehicles that are deployed in wide underwater areas and collect data about salinity, pressure, temperature, speed of current flow, etc. [10, 11]. Then, the collected data are sent to a sink, mostly a navigator, located on the water surface which, in its turn, forward the data to the offshore station for a later analysis and decision making. Unlike the terrestrial sensor communication, the selection of the acoustic communication is due to the multi-path propagation and the strong signal attenuation in underwater environments. Furthermore, the underwater networks provide much more challenges compared to terrestrial ones due to:

- The densely deployment of sensors because of the wide ocean surfaces.

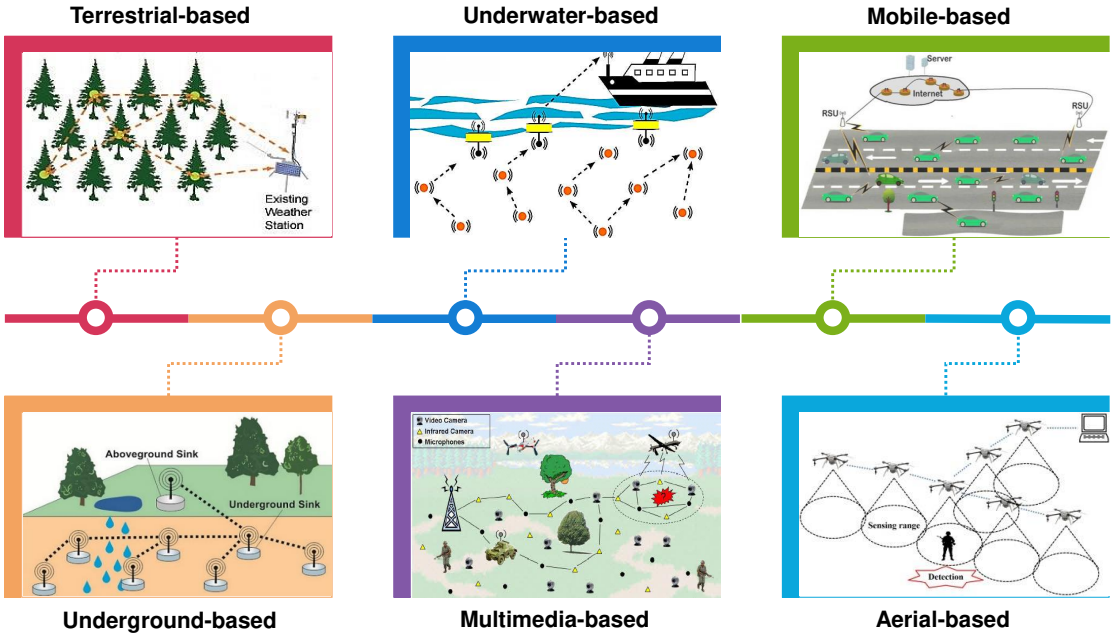



Figure 1.3: Types of sensing-based applications.

- The big data collection resulted from the periodic monitoring of the oceans.
- The energy consumption of acoustic communications is very high and it proportionally increases with the increasing of the amount of data transmission and the distance to the sink.
- The acoustic transmission has a small bandwidth with less reliability and quality of data.

**(IV) Multimedia-based applications:** in this kind of applications, the sensor nodes are relatively inexpensive and provided with cameras and microphones. The node deployment should be planed in advance to satisfy the desired coverage and to ensure a good quality-of-service (QoS) for the end-user. The sensor type used in multimedia application has the ability to store, process and retrieve multimedia data like video, audio, and images. Unfortunately, the deployment of multimedia networks faces more challenges compared to other types of applications. Particularly, the major challenges include the huge amount of processing, compression and bandwidth needed to deal with multimedia data, the high level of energy consumption required during data transmission, and the QoS is not always guaranteed because of the link capacity and delays. Examples of multimedia networks mainly include tracking, security and military applications [12, 13].

**(V) Mobile-based applications:** this type is composed of mobile sensor nodes that have the ability to move autonomously around the target and interact with it. In addition, these nodes can self-organizing and correct their position in the network with their ability for sensing, processing and communication. Unlike static network, the mobile network can maintain a high sensing coverage and connectivity of the monitored area, relocate nodes to fill coverage holes, and reduce the number of sensors required to cover the target zone. However, researchers face many challenges when deploying mobile networks, such as: the self-organization of nodes,

the navigation and control of mobile nodes, and minimizing the node movement in order to save its energy. Common examples of mobile sensor networks are vehicular ad hoc networks (VANET), animal tracking, and search and rescue operations [14, 15].

**(VI) Aerial-based applications:** in this kind of applications, the sensor devices are of low-cost and embedded in unmanned aerial vehicles (UAVs) in order to retrieve information from some inaccessible locations. The UAVs can fly autonomously according to a predefined path, without any human intervention, or be remotely controlled by the end-user. Mostly, the aerial sensor network consists of a set of UAVs with a limited battery power that allow the monitoring of a remote area and sending the collected information over airborne wireless relays to a ground station. Thanks to such network characteristics, we witnessed recently a new revolution in both military and civilian applications including the monitoring of disasters, country borders, and traffic as well as the search and destroy operations, wind estimation, and managing wildfire. However, the deployment of aerial sensor networks faces several challenges such as designing software protocols for autonomous flights, developing routing protocols for wireless communication between UAVs, and hardware failure [16, 17].

## 1.4/ Survey on Sensing-Based Applications

Nowadays, sensing-based networks have a huge number of applications in a wide range of fields. This is due to numerous existing sensor devices (such as thermal, visual, acoustic, magnetic, etc.) where each of them has the ability to monitor one physical condition (such as temperature, humidity, salinity, light, etc.). In this section, we give an overview about well-known fields and real projects in sensing-based applications.

### 1.4.1/ Military Surveillance

Nowadays, the security and safety of people is becoming more and more of a big concern to governments and security forces in general, and human beings and societies in particular. This increasing concern is particularly fueled with the increase in terrorist attacks , and the global increase in the number of related firearm deaths and homicides. Hence, sensing-based technology has taken a great attention from governments and security forces as an efficient solution for defensive and offensive operations. Particularly, sensing technology helps in detecting intrusion, tracking enemy, monitoring battlefield, classifying targets, and detecting nuclear/biological/chemical attacks [18, 19]. Therefore, by deploying low-cost sensor devices, such technology can provide an effective solution that enables the implementation of an automatic activation alarm system after detecting any critical situation. Then, it notifies the police and security forces so they can early intervene and prevent a crime/attack from happening.

### 1.4.2/ Environment Monitoring

Sensing-based technology is broadly used in environmental-based applications for either indoor or outdoor. From one hand, the indoor-based applications aim at monitoring offices and buildings by involving sensor devices such as humidity, temperature, light, air quality, fire, or civil structure deformation. On the other hand, the outdoor-based applications monitor and study various phenomena in order to reduce their effects. Particularly, examples of outdoor applications include the monitoring of traffic, habitat, weather, or the detection of earthquake, seismic and volcano activities. Subsequently, some of the most important real projects in sensor networks include GAEMN (Georgia Automated Environmental Monitoring Network) for environment monitoring [20], Volcano Tungurahua for monitoring volcano activity [21], and GlacsWeb for understanding the relationship between glacier dynamics and climate change [22].

### 1.4.3/ Healthcare Monitoring

Nowadays, hospitals and government departments are struggling to reduce the health costs and improve the service quality. Hospitals rely mainly on nurses who have many duties including caring of patients, communicating with doctors, administering medicine and checking vital signs. Hence, it comes the importance of sensing technology that helps the medical staff in electronic health surveillance of patients with early detection of critical physiological symptoms. Recently, with the rapid spread of COVID-19, sensing technology has taken a great attention from communities and enterprises, and it has significantly reduced the surveillance nurse duties and increased the efficiency of health systems. Mainly, such technology consists of a set of biosensors implanted in and/or on the patient body that constantly monitors their vital signs and sends the collected data toward the medical team for analysis and real-time decision-making [23, 24]. In the commercial market, one can find a huge number of biomedical devices such as epidermal-based, saliva-based, and tear-based biosensors [25]. Such biosensors mostly collect numerical data about patient vital signs (heart rate, respiration rate, oxygen saturation, etc.), image data about patient organs (dental imaging, radiography, cardiology, etc.), or video data for various patient surgery operations (cardiology, ocular or invasive surgery, etc.).

### 1.4.4/ Industrial Monitoring

In the industry sector, sensor technology has been introduced in many applications with the aim to reduce, or even to eliminate the need for human intervention in different industry places, especially for dangerous tasks. For instance, attaching small sensor devices to every machine may help in monitoring its performance without the need for daily visits or automatic check. Sensor technology also reduces the cost associated with using wired solutions for the communication as well as the cost of using insulation to provide the protection from the external conditions that harm the wire physically such as high temperature. In the literature, we can distinguish between two main applications of industrial-based sensing technology: condition monitoring or process automation. From one hand, the condition monitoring includes the surveillance of structural health, such as: building, wind turbine, coalmine, tunnel and bridges [26], or the surveillance of equipment condition (i.e. pipelines and machinery) [27]. On the other hand, the process automation has

potential applications [28] whether for process evaluation (i.e. water consumption, AC energy, and supply/cold chain), or for process improvement (i.e. field irrigation, precision viticulture, HVAC control and production automation). Moreover, industrial-based sensing applications have witnessed the implementation of significant real world projects such as RealFusion [29] for monitoring bulk substances in factory silos, and MCBM (machinery condition-based maintenance) [30] for real time monitoring of machinery spaces using commercially available products.

### 1.4.5/ AGRICULTURAL MONITORING

In various countries, the agriculture forms the backbone of the economic system and represents the main source of livelihood to a large number of the population. However, according to a recent report published by the Food and Agriculture Organization (FAO) of the United Nations [31], the world needs to produce $70\%$ more food in $2050$ than it did in $2006$ due to the exponential increasing of the population number that is estimated to reach $9.7$ billion. Hence, to meet these demands, the integration of sensing technologies in the agriculture operations has pushed this field to the next level and it has been a driving force behind the increasing of the agriculture production at a lower cost [9]. With the help of sensors, the Internet of agriculture things (IoAT) allows the farmers to get live data from anywhere about the environmental and field conditions then to make accurate and quick decisions about every operations of their work, from climate change to precision farming. For instance, with underground wireless soil sensors, the farmers can take better control over the process of irrigation and thus, enhance the water use efficiency. Therefore, the IoAT solutions have gained a lot of attentions from agricultural companies and, as per recent reports [32, 33], the market share is expected to reach $\$15.3$ billion in $2025$ with more than $225$ millions of connected devices. Indeed, such devices can monitor various kinds of agriculture applications such as water and nutrition, crop health, diseases and bug, machinery, etc. and help in several services such as irrigation, pesticides, fertilization, yield condition and storage, etc. [9].

### 1.4.6/ ROBOTS MONITORING

The twentieth century has witness a revolution in the robotic technology sector that highly affects our lives and those of the future generations. Thanks to a combination between sensing and robot technologies, such revolution has led to the emergence of a new generation of robotics called as Modular Robotic System (MRS). Generally, a MRS consists of a set of independent modules, where each of them is equipped with a battery. In addition, each module has the ability to sense the environment, compute the collected data and move on the space according to some degrees of freedom. Thus, the transitions of the module positions allow the MRS to be reconfigured from an initial morphology to the desired one. One of the most significant advantages of the MRS is that it can be programmed to carry out several missions and tasks, which are too complex, dangerous, dirty or boring for humans. Hence, MRS has found its way quickly into a great number of applications including rescue, healthcare, manufacturing, reconnaissance and military missions [34, 35].

### 1.4.7/ WATER AND OCEAN MONITORING

Since ancient time, the oceans have been the center of attention as they cover about the three fourth of the earth surface. According to the United Nations report for oceans [36], $37\%$ of the global population are living in coastal areas, between US $\$3\text{-}6$ trillion/year is the estimated ocean-economy and $2900$ million tons of oil are transported every year by sea. Unfortunately, over the last two decades, the marine life has facing an increasing number of challenges including marine debris, oil spills, loss of biodiversity, ice melting in polar regions, sea level rise, extreme weather events, displacement, etc. In order to study and overcome such challenges, the sensing-based technology has been integrated into the monitoring of the ocean activities. This allows experts to better understand the marine life, help in preserving the natural resources by tracking the pollution and getting an early notification of marine disasters. Indeed, one of the well-known projects in water and ocean monitoring is Argo [37]. Subsequently, ARGO deploys more than $3600$ sensor nodes over the global oceans where each node collects data about salinity, temperature and velocity readings in the upper $2000$ meters of depth. Every ten days, data collected by the sensors are transmitted to a satellite while the nodes are always on the surface.

## 1.5/ NETWORK DESIGN

Transmitting the raw data collected by the sensor nodes to the sink is a fundamental operation in sensing-based networks. Hence, the network architecture plays an important role in the performance of sensor applications. Subsequently, several metrics (such as congestion, energy consumption, network overload, data loss, latency, etc.) are highly affected by the selection of the network architecture. In this thesis, we are interested on the cluster-based network architecture in which the data transmission among sensors and the sink is performed using two-hops communication.

### 1.5.1/ CLUSTER-BASED ARCHITECTURE

Network topology is one of the most key features that should be consider when deploying a sensor network. Although there are many topologies proposed for sensor applications [38], researchers are mainly focused on two architectures: clustering and tree. Indeed, a tree-based sensor network is more suitable for applications requiring a small size of sensors; otherwise, e.g. when the number of sensors gets bigger, the construction of the tree will be very complex. Such reconfiguration of the tree mostly requires a high time processing and network energy consumption especially when a node is failed or its energy is depleted (particularly for those near to the sink). Hence, for less-complexity reason, most of the proposed techniques are dedicated to cluster-based topology in order to maintain the scalability of the network and save its energy. Subsequently, the authors of [38] study the various topologies of sensor network (tree, cluster, chain and flat) while comparing them according to many performance metrics like energy usage, network lifetime, scalability, latency, etc.

In this thesis, we are dedicated to cluster network scheme as an efficient topology to study redundancy among data collected by the sensor nodes. In such topology, the network area is partitioned into subzones called clusters. Inside each cluster, a specific node

called Cluster-Head (CH) is assigned which is responsible to forward data coming from the sensor member cluster to the sink. Mostly, the CH is elected after the network deployment and can be dynamically changed during the network lifetime. It can also be a regular node or a specific more powerful one, depending on the application context and requirements. Figure 1.4 illustrates a two-layer cluster architecture in which the communication among the sensors and their CHs or the CHs and the sink is performed according to a single-hop transmission.

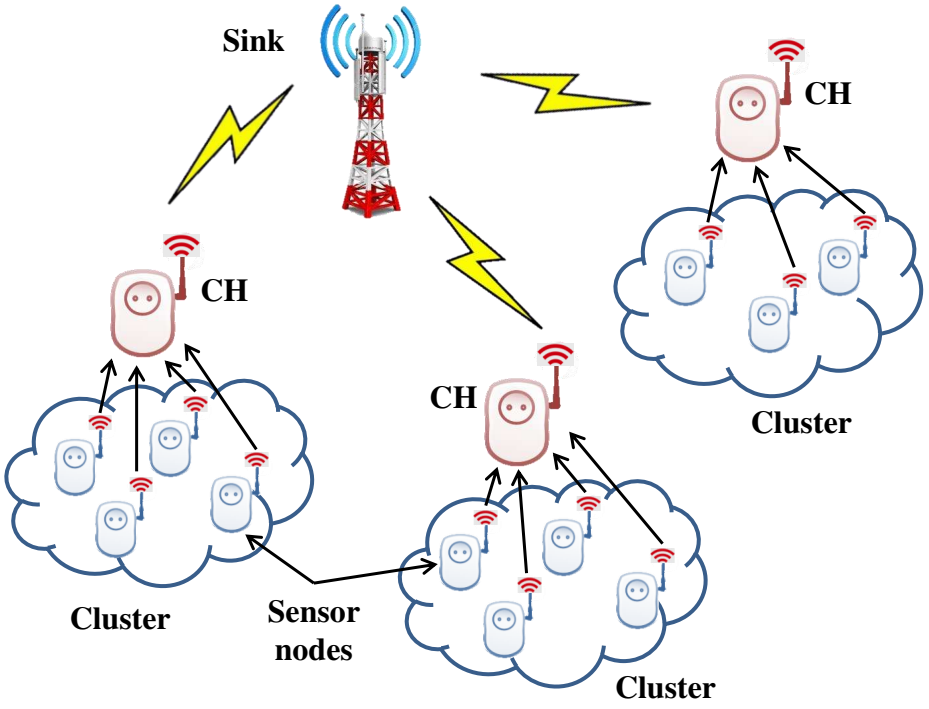



Figure 1.4: Two-layers cluster-based architecture network.

Indeed, dividing network into clusters is not an easy task and it faces many challenges. Hence, one can find a lot of works in the literature that are interested in issues related to cluster network like selection of cluster heads [39, 40, 41], optimization of cluster size [42, 43], communication between sensors/CHs and CHs/sink [44, 45], etc. However, our major concern is to study the variation of data collected by the sensors and not the formation of clusters themselves. Therefore, we consider a geographical clustering scheme in which near sensors are already assigned to the same cluster.

### 1.5.2/ Periodic Data Collection Model

After selecting the appropriate network architecture, the sensor nodes start sensing the surrounding and sending the data toward the sink. Indeed, we can distinguish among three types of data collection in sensor applications: query-based, event-based or periodic-based [46]. In our project, we focus on the last collection model which is used in a large number of applications that require a constant and continuous monitoring, such as: phenomena study, patient observation, habitat surveillance, traffic tracking, etc. In

most of such applications, sensors collect data of interest and forward them to the sink at constant periodic time intervals for analysis and studying purposes.

More formally, let consider a sensor network $N$ consists of $n$ sensor nodes as follows: $N = \{N_1, N_2, \ldots, N_n\}$. Each sensor node $N_i \in N$ may contain $Q$ sensors, e.g. $N_i = \{S_{i1}, S_{i2}, \ldots, S_{iQ}\}$, where each sensor monitors one physical condition. Indeed, in a periodic acquisition model, data are collected on a periodic basis, where each period $p$ is partitioned into time slots. At each slot $t$, each sensor $S_{ij} \in N_i$ captures a new reading $r_t^{ij}$ then it forms, at the end of $p$, a vector of $\mathcal{T}$ readings as follows: $R_{ij}^p = \{r_1^{ij}, r_2^{ij}, \ldots, r_{\mathcal{T}}^{ij}\}$. Therefore, each node $N_i$ will form a matrix $M_i^p$ for all sensor data before sending to its appropriate CH. $M_i^p$ is described as follows:

$$M_i^p = \begin{array}{c} \\ t_1 \\ t_2 \\ \vdots \\ t_{\mathcal{T}} \end{array} \begin{array}{c} \begin{array}{cccc} R_{i1}^p & R_{i2}^p & \ldots & R_{iQ}^p \end{array} \\ \begin{pmatrix} r_1^{i1} & r_1^{i2} & \ldots & r_1^{iQ} \\ r_2^{i1} & r_2^{i2} & \ldots & r_2^{iQ} \\ \vdots & \vdots & \ldots & \vdots \\ r_{\mathcal{T}}^{i1} & r_{\mathcal{T}}^{i2} & \ldots & r_{\mathcal{T}}^{iQ} \end{pmatrix} \end{array} \begin{array}{l} \\ = m_{i1}^p \\ = m_{i2}^p \\ \vdots \\ = m_{i\mathcal{T}}^p \end{array} \tag{1.1}$$

Furthermore, we also consider that a set $m_{it}^p = \{r_t^{i1}, r_t^{i2}, \ldots, r_t^{iQ}\}$ represents the readings collected by all the sensors of $N_i$ during the slot $t$. Thus, the data collected by the node $N_i$ during the period $p$ can be converted into: $M_i^p = \{m_{i1}^p, m_{i2}^p, \ldots, m_{i\mathcal{T}}^p\}$.

## 1.6/ Sensor Network Challenges

Despite sensor network is a promising technology for a wide range of problems, researchers have to face many challenges related to the design of sensing-based applications. Some fundamental challenges are related to the network deployment and configuration (such as routing, coverage, scalability, etc.) while other ones are related to the limited sensor resources (such as energy and short communication range), and, finally, other challenges are related to the data management in sensor network (such as big data collection, data latency and accuracy, etc.). In the next sections, we describe some of these issues and challenges.

### 1.6.1/ Deployment

The sensor deployment represents the first challenge for the end-user, since it is considered as the first fundamental phase in the network lifecycle. Subsequently, the network deployment represents the way in which the sensors are deployed and arranged in the application area. Hence, the deployment task is very demanding in terms of planning, design and implementation. Mainly, we can distinguish between two categories of sensor deployment: deterministic or randomly. From one hand, the deterministic deployment is mostly performed through a human or robot where the sensors are deployed in a deterministic way in predefined locations. On the other hand, the random deployment is adapted to remote, harsh or inaccessible terrains, where the sensors are distributed from a plane or a rocket. Indeed, the random deployment offers a practical solution in terms

of time and cost, however it provides more challenges comparing to the deterministic deployment [47, 48, 49].

### 1.6.2/ Energy Consumption

Minimizing the energy consumption in sensors is a major challenge in the design of hardware, software, communication and deployment of sensor network. From one hand, the sensor energy conservation leads to prolong the network lifetime and ensures a long-time monitoring of the target zone. On the other hand, it reduces the cost of the network deployment and the reconfiguration time in case of energy depletion of some sensors after network deployment. Furthermore, recharging or replacing sensor batteries is not always possible in sensor networks especially in harsh, inaccessible or dangerous zones. Hence, researcher and market communities have put great efforts to optimize the energy consumption in sensor network [50]. To reach their goal, different approaches have been investigated, such as: 1) Power supply technologies that allow to extend the capacity of the sensor batteries by integrating low-cost and renewable energy sources, such as: solar panel and combustible battery. 2) Hardware energy optimization that aims to reduce the energy consumption of various components of the sensor, especially processing and communication modules. 3) Efficient communication protocols that aims to reduce the amount of data transmitted in the network while maintaining the integrity of the sent information and the quality of service. Indeed, data communication is considered as the most energy-consumed operation compared to other operations, particularly processing and sensing.

### 1.6.3/ Security

Sensor networks are mostly considered as critical systems in many applications such as hospitals, airports, battlefield, industrial plants, and others that require a high level of security against attackers. We distinguish between two types of security when deploying a sensor network: physical and logical. From one hand, sensor devices are mostly deployed in non-secure areas, especially in the cases of random deployment and distribution, and can be subject to attack by intruders, e.g. physical attacks. Indeed, sensor degradation and communication might be caused by humans or animals, as well as, the environment and disasters, such as: bad weather, rain, fire, inundation, etc. On the other hand, sensor networks are subject to information-based attacks, or logical attacks, due to the wireless communication and less-infrastructure characteristics of such network. Examples of logical attacks include tempering, black hole, selective forwarding, Sybil, jamming, exhaustion and others [51]. Therefore, ensuring a high level of confidentiality, integrity and reliability of data routed from sensors to the sink becomes essential to guarantee the logical security of the sensor network.

### 1.6.4/ Scalability

Sometimes, sensor networks are consisting of a significant number of sensor nodes (reaching thousands or even hundreds of thousands) depending on the monitored zone area and the application requirements. In addition, end-users are enforced in many applications to increase the scalability of the network after sensor deployment for data relia-

bility purpose. Hence, the scalability is becoming a challenging issue in sensor network, that has to keep a high performance level when the number of nodes increases. Thus, it is important to design and develop new routing protocols to cope with changing topology of the sensor network as well as to reduce the packet collision and eliminate the redundant data circulated on the network [52].

### 1.6.5/ ROUTING

Routing is considered as one of the important challenges in sensor network that needs to be treated efficiently. Indeed, a routing protocol is defined as the best path to send the data from the source, e.g. the sensor nodes, to the destination, e.g. the sink. However, there are several issues that should be taken into consideration when designing a routing protocol [53]:

- The selection of the next-hop which is crucial to determine the network overhead, the data latency, and the overall routing energy efficiency.
- The energy balancing where the routing protocol should balance between the sensor data transmission and its consumed energy; more data are transmitted more the energy consumption is, and vice versa. Hence, the shortest path is not always the best energy saving solution since it leads to quickly deplete the energy of the sensor nodes close to the sink. Thus, an efficient routing protocol must find several paths to the sink in order to balance the energy in the network.
- The data latency where, in some applications, data delivery time represents a critical factor for decision makers and any delay in receiving the data might affect the quality of the decision. For instance, a late decision in healthcare applications can lead to the death of a patient while, in military applications, can lead to a loss of a vehicle tracking or in worse case scenario results in injury or life loss
- Data reliability which indicates the accuracy of data received by the end-user. Thus, a routing protocol should be designed in a way that reduces the packet loss or distortion in the network.

### 1.6.6/ COVERAGE

Ensuring a maximal coverage of the monitored zone becomes an important issue when deploying a sensor network. Mainly, the coverage zone ratio is determined according to the number of deployed sensor nodes and the sensing range of each one. Thus, in order to ensure a full coverage of the monitored zone along the network lifetime, it is important to save the sensor energies as long as possible. Indeed, some sensor-based applications, such as environment-based and underwater-based, allow a certain level of flexibility regarding the network coverage ratio while another applications, such as healthcare-based and military-based, require a hard constraint regarding the covered zone. Therefore, researchers have focused on scheduling strategies that aim to select a subset of sensor nodes to be active during a period while switching-off other ones into a sleep mode. However, the selected sensor nodes should be selected in a way that the energy consumption in the sensor nodes is enhanced and an accepted level of coverage zone is ensured [54].

### 1.6.7/ Big Data Collection and Management

Due to the huge amount of data collected, the periodic model provides a significant redundancy in sensing-based applications. This redundancy might occur whether at the sensor node or CH levels. Consequently, such redundancy among data will lead, from one hand, to complicate the data analysis at the end-user and, from the other hand, to deplete the limited energy of the network nodes. Hence, data reduction approach is at the heart of research focuses nowadays as an efficient way to handle big data collection, eliminate the data redundancy, reduce the data transmission, saving the network energy and improve the decision-making in sensor networks. In the next sections, we further detail the problem of data redundancy and possible solutions at each network level.

#### 1.6.7.1/ At Sensor Node

Obviously, the use of periodic monitoring model produces a huge amount of data collection that consume most of the sensor energy during their transmission. Therefore, in order to conserve its energy and prolong its lifetime, the amount of data transmission from each sensor should be reduced without affecting the collected information. This can be performed by eliminating the redundancy among the collected data either within the same period, e.g. on-period, or among successive periods, e.g. in-period. From one hand, on-period redundancy happens due to the slow variation of the monitored condition or when a small value is assigned to the slot time. This leads to increase the similarity among readings collected in each period and, consequently, it increases the redundancy among the data transmitted from the sensor. In order to eliminate on-period redundancy, researchers have proposed several approaches such as aggregation, compression or prediction. On the other hand, the redundancy level among the collected data highly depends on the variation of the monitored condition. For instance, the monitoring of humidity or temperature will produce a high redundancy level because such conditions are slowly varying during the progress of periods. Thus, in-period data redundancy should be also eliminated in order to further conserve the sensor energy. In the literature, one can find several approaches that allows eliminating the in-period data redundancy in sensor networks, particularly adapting sensing frequency and on-off transmission.

#### 1.6.7.2/ At CH Node

As already mentioned, the sensor nodes are mostly scattered in a random way over the monitored areas. This leads to a high level of spatial-temporal correlations between sensor nodes. Thus, when receiving the data sets from all sensors at the end of each period, the CH can benefit from such correlations in order to eliminate the data redundancy among neighboring sensors, e.g. in-node data redundancy, before sending them toward the sink. Therefore, the periodic data transmitted by the CH will be reduced which will save its energy and facilitate the data analysis task of the end-user. In the literature, researchers proposed many approaches to eliminate in-node data redundancy such as data clustering, in-network data aggregation, data fusion, and sensor scheduling.

#### 1.6.7.3/ At Sink

After receiving data from all CHs, the sink applies some preprocessing techniques before sending them the end-user to make a decision. Examples of preprocessing techniques include the estimation of missed/lost readings, the regeneration of predicted raw data, and the data fault detection, which can highly affect the decision made by the end-user. Hence, the relevance of any decision-making system will be mostly related to the quality of collected data and the application itself where one decision model cannot fit all sensor applications. Therefore, one can find many decision-making systems based on fuzzy logic and temporal decision, multi-criteria and multi-agents models that are mostly dedicated to a specific sensor application. However, such efforts are embarrassing and it becomes essential to investigate more in developing new models and systems for decision-making in sensor networks.

## 1.7/ Energy-Efficiency and Data Reduction: A Background

Big data management is a challenging process in sensing-based applications as data are mostly correlated and contains a high level of redundancy. Thus, what to keep or discard becomes a crucial task affecting the accuracy of the collected data thus the decision made at the sink. Current research on big data management in sensor networks is focused on redundancy reduction methods with the aim to reduce the amount of sensor data collection thus, saving the network energy and enhancing the decision-making [55, 56]. The objective of such methods is to study the similarities among the collected data, eliminate the existing redundancy and deliver a useful information to the end-user in order to make a suitable decision. Subsequently, the redundancy reduction is either applied at the raw source, e.g. sensor level itself, or at intermediate nodes, e.g. the CHs, while the decision-making process is performed at the sink. In this section, we present a state-of-the-art for energy-efficiency and data reduction in sensor networks while classifying the proposed techniques based on aggregation, compression, prediction, clustering, in-network aggregation, adapting sensing frequency, sensor scheduling or decision-making.

### 1.7.1/ Aggregation-based Techniques

By definition, data aggregation is the process of gathering similar data collected by each node into one useful information thus, the redundancies are eliminated and the integrity is preserved. In the literature, researchers have been proposed a significant number of data aggregation techniques in order to save the sensor node energies and prolonging the network lifetime [57]. The authors of [58] propose an entropy-driven data aggregation with a gradient distribution (EDAGD) technique that is relying on three algorithms. The first algorithm is called a multi-hop tree-based data aggregation and aims to reduce the transmission distance between the sensors and the sink by minimizing the number of hops required to reach the destination. The second algorithm is a tree-based aggregation scheme that uses the entropy and the Choquet integral that allows to monitor and detect abnormal events based on the sleep/active nodes strategy. The last aggregation method is a gradient deployment algorithm which aims to deal with the energy hole problem in IoT

applications. In [59], the authors propose a Priority-based Compressed Data Aggregation (PCDA) technique in order to reduce the amount of heath data transmitted. PCDA uses compressed sensing approach, based on the sensing matrix and convex optimization, followed by a cryptographic hash algorithm, which uses a key pre-distribution scheme, at the biosensor level to save information accuracy before sending data for diagnosis. The simulation shows that PCDA ensures a low execution time and communication overhead with a moderate energy consumption. In [60], a mechanism based on the collaboration between unmanned aerial vehicles (UAV) and sensor networks for crop monitoring in precision agriculture is proposed. The data collection scheme in the proposed mechanism is performing according to the following steps: first, a cluster-based scheme is proposed to group neighboring sensors into clusters and assign a cluster-head for each cluster. Second, a data aggregation method based on the minimum and maximum values extraction is applied at each sensor to reduce its data transmission to the CH. Third, a path planning strategy is designed to collect data from the CHs. The last step introduces an edge-fog-cloud computing algorithm to process data at the sink node. In [61], a two- level node mechanism has been proposed which is dedicated to periodic sensor applications. First, the authors propose an on-node aggregation method to remove redundant data collected by the sensor. Then, an in-node data reduction called prefix frequency filtering (PFF) is introduced at the CH level. PFF allows CHs to find similarities among data collected by neighboring nodes in the same cluster, using Jaccard similarity function. The authors of [62, 63] propose two data aggregation schemes, namely block diagonal matrix and block upper triangular matrix, for cluster-based UASNs inspired by the Distributed Compressed Sensing (DCS) technique. The main objective of such schemes is to generate RIP-preserving (Restricted Isometric Property) measurements of sensor readings by taking multi-hop underwater acoustic communication cost into account. Finally, a distributed compressed sensing reconstruction algorithm, called DCS-SOMP, is adopted to recover raw sensor readings at the fusion center. In [64], a semi-structured protocol based on the multi-objective tree is proposed, in order to reduce transmission delays and enhance the aggregation probability. In such a work, the routing scheme explores the optimal structure by using the Ant Colony Optimization (ACO). In [65], the authors propose a Cycle-Based Data Aggregation Scheme (CBDAS) in order to reduce the amount of data transmitted to the base station (BS). In CBDAS, the network is divided into a grid of cells, each with a head. The network lifetime is prolonged by linking all cell heads together to form a cyclic chain, where the gathered data move from node to node along the chain, getting aggregated. The authors of [66] propose a Semi Distributed Heuristic Energy efficient Aggregation Tree (SDHEAT) algorithm for WSN. Mainly, SDHEAT is based on three concepts: heuristic tree formation, sensing priority and distributed nature and aims to reduce the overall network consumption while conserving information integrity. In [67], the authors propose a multidimensional and multidirectional data aggregation (MMDA) technique in order to enhance the data communication and ensure the privacy of the data. MMDA allows each IoT device to organize the data into matrices then applying an aggregation process in two directions, e.g. rows and columns.

### 1.7.2/ COMPRESSION-BASED TECHNIQUES

Data compression aims to modify, encode, or convert the bits structure of data in order to reduce the size of storage or transmission of such data. Hence, the data compression has taken a great attention from researchers as an efficient solution for bit-rate reduction

and energy saving in sensor networks [68]. The authors of [69] propose a control scheme based on data compression and sensing rate in order to reduce the amount of data collected at the sink node. The idea behind this scheme is that every parent node sends a threshold, called data quota, to all its node children. According to the received quota and its remaining energy, the children node selects its suitable compression method and its sensing rate during the next period. In [70], an efficient and robust compression method is proposed, Sequential Lossless Entropy Compression (S-LEC). S-LEC uses a differential predictor that arranges the alphabet of integer residues into a number of groups. Subsequently, S-LEC assigns two codes to each group: entropy code and binary code. The first code specifies the group where the second one represents the index inside the group. In [71], a coding provenance scheme (CBP) has been proposed. Compared to traditional compression techniques, CBP ensures a high provenance compression rate as well as it encodes and decodes incrementally the compression ratio at the base station depending on the condition observed. The authors of [72] propose a compressed data reduction technique dedicated to underwater sensing applications. The proposed technique is consisting of two layers: compressed sampling and data reduction. After forming clusters, the first layer randomly selects a number of nodes for conducting sampling. Whilst, the second layer proposes a full sampling technique in order to minimize the entire energy consumed during data transmission. In [73], the proposed model uses spatial node clustering as well as the principal component analysis (PCA) in order to compress the collected data. In a first step, the authors group sensors with a strong correlation into clusters using novel similarity metrics like magnitude and trend. Then, the authors propose an adaptive strategy for the selection of cluster heads. Lastly, PCA is applied at the cluster heads with a predefined compression error in order to maintain the variance the collected data. Finally, the selected cluster heads apply PCA with an error bound guarantee to compress the data and retain the definite variance at the same time. In [74, 75], the data compression and encryption are combined together in order to keep secure data after compressed and before sending them. First, in [74], the authors propose a Fuzzy-transform (F-transform) compression method based on the discrete wavelet transform model. Then, in [75], an encryption layer called B-spline is added in order to encrypt data before sending to the sink. The authors of [76] proposed a data reduction technique dedicated to wireless seizure systems. In addition to local compressive sensing, the proposed technique selects a set of features, specifically those with nonlinear autocorrelation, to reduce the seizure signals sent to the data server.

### 1.7.3/ PREDICTION-BASED TECHNIQUES

The idea behind data prediction is to build, based on the collected data, a predictive model in order to send it to the sink which, in its turn, regenerates the raw data [77]. Researchers of [5] have presented a review article about various data prediction mechanisms proposed at the literature for sensor networks, while comparing the difference between them. The authors of [78] propose an adapted version of the dual prediction scheme (DPS) algorithm. The new version uses a collection of models for data prediction during the past sequences of the DPS algorithm, without updating the history data table classically. Indeed, the new prediction model is computed at the sensors and sent to the sink or vice-versa. The performance of DPS is tested using the data collected from the meteorological station located at Tlemcen (Algeria) while the results show that the data transmission ratio is reduced by more than $90\%$ when accurate predictions are achieved.

In [79], the authors propose an AUV-aided solution called a prediction-based delay optimization data collection (PDO-DC) algorithm aiming to reduce the data collection delay in acoustic underwater IoT (AUIoT). First, PDO-DC uses a machine learning technique called Kernel Ridge Regression in order to build and update the prediction to fit the collected data. Then, it proposes an AUV path planning strategy based on the competition coefficient in order to reduce the number of visited nodes when collecting the data and thus, reduce the collection delay and avoid the packet loss. In [80], an AgriPrediction framework, which combines between LoRa IoT technology and autoregressive integrated moving average (ARIMA) prediction, is proposed. AgriPrediction builds a prediction engine that aims to avoid potential crop failure proactively and notify the farmer, through short and medium communication, for remedial actions as soon as possible. The authors of [81] propose a hybrid prediction model based on two algorithms; A stagewise algorithm applied at sensor level uses a set of data points to build a predictive model to reduce sensor data transmission. Whilst, the other algorithm is used by the sink node to reconstruct the raw data generated by the sensors. In [82], a vector-based model for predicting sensor readings is proposed. After considering a linear distribution of data, the authors search for the correlation between the data using a line equation through two vectors in a n-dimensional space. The authors of [83] propose a data approximation mechanism for temporal readings collected by each sensor. The mechanism converts original readings into binary codes then an application layer is implemented in order to send the converted data. The authors of [84] propose an unsupervised machine learning algorithm, called Kohonen, for predicting data generated by the sensors. Kohonen introduces a self-organizing map based on a predictive temporal model that makes sensor in standby mode to reduce its transmission. The authors of [85] propose an Adams-Bashforth-Moulton algorithm that aims to optimize the accuracy of prediction obtained with Milne Simpson algorithm proposed in [86]. Both algorithms are simulated on real data sensor and an optimization level of energy and accuracy is noticed. In [87], the authors propose a polynomial regression-based data aggregation protocol that conserve network energies as well as the privacy of sensed data. Instead of sending its raw data, each sensor uses coefficient regression polynomials to represent their data while the aggregation is made on such secret coefficients. In [88], the authors propose a mechanism that predicts future values based on the past one. The mechanism uses an autoregressive model of order $p$ and allows to study the variation in sensed data along with the network lifetime. In [89], a derivative-based prediction (DBP) technique is proposed. DBP is dedicated to WSN applications requiring high data accuracy and it predicts the variation of data collected by a sensor node. In [90], an online data tracking and estimation (ODTE) is proposed in order to tracking poor data collected at the sink. ODTE is mainly based on two systems: Data prediction system (DPS) and distortion factor (DF). DPS is used at the sensor in order to reduce its transmission using a defined limit while DF estimates an optimal data collected at sink node.

### 1.7.4/ CLUSTERING-BASED TECHNIQUES

Data clustering is the process of grouping similar data into clusters then to eliminate the redundancies existing inside each cluster [91]. In [92], the authors propose a layered adaptive compression design for efficient data collection (LACD-EDC) in industrial sensor network. LACD-EDC is based on the clustering data scheme and searches the spatio-temporal correlation within (e.g. intra) and among (e.g. inter) clusters. Then, a

compression method is proposed at the sensor level followed by a recover technique at the sink in order to regenerate the raw data and achieve an approximate data collection. The authors of [93] propose a cluster-based data gathering algorithm for sensor network called lifetime-enhancing cooperative data gathering and relaying (LCDGRA). Basically, LCDGRA works on three phases: the first phase groups the sensor nodes into clusters based on K-means clustering while applying a compression technique, e.g. Huffman coding algorithms, in each cluster. The second phase assigns a set of relay nodes to each CH in order to aggregate data before sending to the sink node. In the last phase, the aggregated data are coded based on random linear coding and then relayed to the base station. In [94], an energy-efficient adaptive clustering routing algorithm (ACUN) for AUIoT has been proposed. ACUN optimizes the lifetime of the cluster-heads by integrating a selection method based on the distance between CHs and the sink, the residual energy of the CHs and the size of the competitive radius. Accordingly, ACUN adopts a set of routing rules, either single-hop or multi-hop, in order to balance the energies of the nodes. The authors of [95] introduce a fuzzy clustering scheme based on particle swarm optimization that increases the AUIoT network lifetime. The proposed scheme designs a fitness function to select the CHs of clusters based on the remaining node energies and the communication range between nodes-CHs and CHs-sink. The authors of [96] propose a fault-tolerant multipath algorithm for cluster-based WSN applications. The algorithm consists of three steps. The first step allows the formation of clusters along with a majority voting technique at the CH to detect the fault of nodes; the second step selects a backup node, from each cluster, to store the data of the whole cluster in order to increase the fault tolerance of the corresponding CH. The last step allows each node to select three paths based on several metrics to send the data toward the sink. In [97], the authors propose an energy efficient routing protocol in order to provide data to the irrigation system. The proposed protocol works in three phases. The first phase aims to divide the monitored area into terrains. The second one selects a cluster-head for each terrain using the fuzzy rules adapted to the remaining sensor energies and the distance to the sink. The last phase selects a set of relay nodes to perform data transmission between nodes and sink. The authors of [98] propose a routing protocol called Gateway Clustering Energy-Efficient Centroid (GCEEC) for WSN. The objective of GCEEC is to improve the load among the sensor nodes as well as selects and rotates the CH near the energy centroid position of the cluster. The results show that GCEEC can highly extend the network lifetime and reduce the network overload; however this is limited to many assumptions taken during the tested scenario. The authors of [99] propose a data aggregation clustering scheme in order to reduce the transmission of redundant data in AUIoTs. The proposed scheme works in rounds where each round consisting in four main phases: initialization, cluster-head selection, clustering, and data aggregation. In [100], the authors propose EBDSC, a distributed Energy-Balanced Dominating Set-based Clustering scheme, to extend the network lifetime by balancing energy consumption among different nodes. In EBDSC, a node becomes a cluster head candidate if it has the longest lifetime among its neighbors. In [101], a Distributed K-mean Clustering (DKC) method has been proposed for WSN. The idea behind DKC is to aggregate data based on the adaptive weighted allocation. DKC algorithm tries to eliminate data redundancy as much as closer to the sensor nodes in order to avoid the overloading of the network. In [102], a semantic clustering technique is proposed to group sensor nodes into clusters according to semantic information and to the network connectivity. It consists in comparing the query sent by the sink and the collected data. Once a sensor node finds that its data satisfies the query, it selects itself as cluster head (CH) and starts forming the semantic

cluster with the nodes whose data also satisfy the same query. This approach is suitable for the data aggregation of in-network query type. In [103], the authors propose a data aggregation scheme named DMLDA, Dynamical Message List based Data Aggregation, based on clustering routing algorithm. DMLDA mainly defines a special list structure to store history messages, which is used to evaluate the message redundancy instead of the period delay. Another semantic clustering method based on fuzzy system was proposed in [104] to find out the semantic neighborhood relationship. It is an event based clustering approach. It considers two kinds of clustering, physical and semantic. The physical clustering groups the nodes into clusters based on a hierarchical organization composed of two levels. The first one contains the CHs while the second one the members (e.g. sensor nodes). When the data collected by the sensor node matches with the monitored event, it becomes candidate. However, on the other hand when the data changes, it becomes a semantic neighbor. Then, the CHs use a fuzzy inference system and exploit the data of all the semantic neighbors which are in the same cluster to obtain an aggregated data.

### 1.7.5/ IN-NETWORK BASED TECHNIQUES

The in-network data approach is used at an intermediate node, mostly called aggregator or Cluster-Head (CH), and aims to a find correlation between neighboring nodes so as to transfer valuable data to the sink. In [105], the authors propose an energy-efficient communication method dedicated to periodic underwater sensor applications. On the basis of the proposed technique, each node cleans its collected data before transmitting to the appropriate CH. When receiving datasets, the CH applies K-means algorithm adopted to the ANalysis Of VAriance (ANOVA) with statistical tests in order to eliminate inter-node correlations. In [106], the authors are dedicated to reduce the data transmission at the CH under a cluster-based underwater network. The proposed technique uses two distance functions, e.g. Euclidean and Cosine, in order to search the data correlation among neighboring nodes, thus removing the data redundancy, before sending the data to the sink. The authors of [107] propose two data filtering approaches to improve energy efficiency on agricultural WSNs. The first approach filters the data collected at the sensor node using a simple moving average (SMA) method. The second approach is dedicated to nodes with one sensor board and it uses the Threshold Sensitive Energy Efficiency Sensor Network (TEEN) protocol. The authors of [108] propose a supervised linear dimensionality (LDR) reduction technique to reduce the dimensionality of the original data to such that it is well-primed for Bayesian classification. This is done by sequentially constructing linear classifiers that minimize the Bayes error via a gradient descent procedure, under an assumption of normality. In [109], the authors propose an aggregation and transmission protocol (ATP) based on clustering approach to conserve energy in periodic sensor networks (PSNs). Instead of sending raw data to the CH, ATP allows each sensor to eliminate redundancy among its collected data and to adapt its data transmission to the CH, using one way Anova model and Fisher test. In [110], the authors propose a data management framework for data collection and decision making in connected healthcare. The framework relies on three algorithms: first, an emergency detection algorithm sends critical records directly to the coordinator; second, an adaptive sampling rate algorithm based on ANOVA and Fisher test allowing each sensor to adapt its sampling frequency to the variation of the patient situation; third, a data fusion and decision making model is proposed at the coordinator and it is based on a decision matrix and the fuzzy set theory.

In [111], a two-level node mechanism has been proposed to periodic sensor applications. First, the authors propose an on-node aggregation method to remove redundant data collected by the sensor. Then, an in-network data reduction is introduced at the CH level that allows CHs to find similarities between data collected by neighboring nodes in the same cluster, using similarity functions. Then, several versions of that approach, in [46] and [112], have been proposed in order to optimize the data latency at the CH level. The authors of [113] propose a Semi Distributed Heuristic Energy efficient Aggregation Tree (SDHEAT) algorithm for WSN. Mainly, SDHEAT is based on three concepts: heuristic tree formation, sensing priority and distributed nature and aims to reduce the overall network consumption while conserving information integrity.

### 1.7.6/ Adapting-based Sensing Frequency Techniques

The main objective of sensing frequency adaptation is to adjust the sensor sampling according to the variation of the monitored condition. This will lead to only collect the necessary data while preventing the collection of redundant one during the collection. The authors of [114] propose an Adaptive Sampling Approach to Data Collection (ASAP) which splits the network into clusters. A cluster formation phase is performed to elect cluster heads and select which nodes belong to a given cluster. The metrics used to group nodes within the same cluster include the similarity of sensor readings and the hop count. Then, not all nodes in a cluster are required to sample the environment. In [115], the adaptation of sampling rate of the sensor node is based on system-context and application-context levels. On one hand, the availability of harvesting energy represents the system-context to identify the maximum rate of sampling to be assigned to the sensor node. On the other hand, the user request represents the application-context, where a feedback from a system executing specific rules of user or field scientists is used to set the rates of sensor node sampling in optimal way. The authors of [116] propose an efficient adaptive sampling approach based on the dependence of the conditional variance on measurement variations over time to allow each sensor node to adapt its sampling rate to the physical changing dynamics. In [117], the authors propose two sampling rate adaptation techniques: exponential double smoothing adaptive sampling (EDSAS) and Wiener filter based adaptive sampling (WFAS). Both algorithms search the correlation between current and previous collected data and aims to minimize the sensor sampling rate while a high level of data accuracy. In [118], a centralized adaptive method is proposed and the sampling rate is derived based on a Kalman filter. In such method, the sampling rates of the sensor nodes are established by the sink. The authors of [119] define a spatial Correlation based Collaborative MAC protocol (CC-MAC) that regulates sensor node transmissions so as to minimize the number of reporting nodes while achieving the desired level of distortion. The authors of [120] propose three mechanisms that allow the sensor to adapt its sampling rate to the variation of the monitored environment. The proposed mechanisms are respectively based on similarity functions (Jaccard coefficient), distance functions (Euclidean distance) and analysis variance with statistical tests (ANOVA and Bartlett test). The proposed techniques work on rounds, where each round consists of a set of period time, in which the sensor adapts its sampling frequency at the end of each round. By adapting different scenarios, the proposed techniques realize the minimum energy consumption with accurate data collection. In [121], the authors propose a TA-PDC-MAC protocol, a traffic adaptive periodic data collection MAC, which is designed in a TDMA fashion. This proposed protocol is designed in the way that it as-

signs the time slots for nodes activity due to their sampling rates in a collision avoidance manner. The authors of [122] propose a district partition-based data collection algorithm with an event dynamic competition in acoustic underwater IoT. The proposed algorithm defines a metric called value of information (VoI) that determines the priority of the packet transmitted from each node. Then, the whole network is divided into subregions and an Q learning algorithm of reinforcement learning is proposed in order to determine the path of the AUV in each subregion. In [123], the authors introduce new atmospheric sensors for measuring air and soil moisture of an agricultural field. Then, a region-based routing algorithm is proposed that allows an efficient data collection from the sensors according to two metrics: residual energy and distance between nodes. In [124], the authors propose an energy-efficient adaptive sampling mechanism which employs spatio-temporal correlation among sensor nodes and their readings. The main idea is to carefully select a dynamically changing subset of sensor nodes to sample and transmit their data. In [125], a machine learning architecture for context awareness is used which is designed to balance the sampling rates (and hence energy consumption) of individual sensors with the significance of the input from that sensor. The authors of [126] propose an adaptive energy aware quality of service (AEA-QoS) algorithm in order to ensure a reliable data delivery in underwater WSN. AES-QoS is two-fold: first, it uses a discrete time stochastic control process and deep learning techniques in order to control the data transmission to the sink. Second, it selects a set of nodes to perform data transmission while optimizing the reliability of communication link, the energy consumption and the propagation delay.

### 1.7.7/ SCHEDULING-BASED TECHNIQUES

The network scheduling is the process of searching the correlated nodes then to select a subset of those having strong correlation to be in active mode while switching-off the others into sleep mode [127]. In [128], the authors propose a centralized algorithm design and an optimizing protocol for scheduling the sensors during a specified network lifetime. The objective is to maximize the spatial-temporal coverage by scheduling sensors activity after they have been deployed. The authors of [129] propose a structure fidelity data collection (SFDC) technique dedicated to the cluster-based periodic applications in WSNs. SFDC searches both spatial and temporal correlation between nodes, using distance functions and similarity metrics respectively. Then, it exploits the dependencies to reduce the number of nodes required to work for sampling and data transmission and prove that such reduction is bound to save energy. The authors of [130] propose a spatial-temporal model to extend the network lifetime based on three similarity metrics: Euclidean Distance, Cosine Distance and Pearson Product-Moment Coefficient (PPMC). Then, they propose a scheduling algorithm for switching correlated sensor nodes to the sleep mode. By performing real experiments, the authors show that PPMC gives the best results, in terms of conserving network energy, compared to other similarity metrics. The authors of [131] propose an energy efficient mechanism for wireless body sensor network based on a sleep scheduling strategy and dominating set method. After constructing the dominating graph, the sink selects, based on two approximation algorithms and a polymatroid function, a subset of nodes to collect the data (e.g. active nodes) while switching the other nodes to sleep mode. In [132], a priority-based energy harvesting scheme for charging embedded sensor nodes in wireless body sensor networks (WBSN) has been proposed. The proposed scheme uses the CSMA/CA protocol in order to switch power from the primary unit to the secondary unit thus, saving the sensor voltage level and reducing the

transmission losses. In [133], the authors propose an Efficient Data Collection Aware of Spatial-Temporal Correlation (EAST) for energy-aware data forwarding in WSNs. In EAST, nodes that detected the same event are dynamically grouped in correlated regions and a representative node is selected at each correlation region for observing the phenomenon, while the other nodes are switched to sleep mode. The authors of [134] propose a multidimensional behavioral clustering that uses Pearson correlation as well as the linear regression in order to reduce the communication activity of sensors. Then, the authors introduces two methods, fractal clustering multidimensional WSN (FCM) and similarity measure in multidimensional WSN (SMM), in order to maintain the cluster topology of the network.

### 1.7.8/ DECISION-MAKING BASED TECHNIQUES

In the literature, we can find several decision-making models proposed to help end-user to take the suitable action for a given application. The authors of [135] propose a multilevel data fusion architecture called Hydra composed of three layers. The low-level phase that introduces a data fusion method at the sensor node; the medium-level phase that alerts the farmer about a set of predefined events, when they occur and the high-level phase that uses a decision fusion technique for multiple data applications. Furthermore, Hydra is developed for two applications: soil moisture monitoring and plant water evaporation. In [136], a modern platform for healthcare information systems consisting of three layers is proposed. The first layer is composed of various data health sources such as sensors, clinical report, medication, etc. The second layer processes and store data and it uses various Hadoop tools including Sqoop, HDFS, HBase, MapReduce, and Hive. The last layer is responsible for applying business intelligence (BI) solutions over the stored data and it uses SpagoBI tools as an open source BI suite. The authors of [137] propose a decision-making system based on a selection sensor mechanism for monitoring soil temperature, humidity and air-and-water quality. First, the system defines the optimal number of sensors needed to monitor the zone then it introduces two methods, e.g. $a(t, n)$ and agronomy function, to assess the plant growth and production yield rates respectively. In [138], a cloud-based connected healthcare system, called BigReduce, is proposed. The objective of BigReduce is to minimize the data processing cost at the base station according to two schemes applied locally at the IoT sensors: reduction and decision schemes. The authors of [139] propose a deep learning mechanism based on the fractional cat-based swarm algorithm for patient situation's assessment and decision-making. First, the nodes are organized into clusters where a cluster-head (CH) is selected for each cluster based on the harmony search algorithm and a particular swarm optimization. Then, the CH receives the records from the nodes and classifies them based on the belief network in order to detect the emergency situations. In [140], a framework for a stress detection and evaluation has been proposed. The framework works by detecting first stress signals according to skin conductance parameter, then the stress level is evaluated through a fuzzy inference system based on patient vital signs, particularly heart rate, respiration rate and average blood pressure. The authors of [141] propose a multi-sensor fusion and decision-making mechanism for patient monitoring through WBSN. The objective of the proposed system is to detect gait abnormality in subjects with neurological disorders based on the gait features (especially spatio-temporal correlation, gait asymmetry and regularity) and machine learning approach. In [142], the authors propose a framework that integrates both IoT and cloud to increase the productivity of the crops in the agricul-

ture fields. The framework provides a real-time analysis of the collected data placed in crops and helps the farmer to reduce its time and its energy when monitoring the crop growth. The authors of [143] present a new paradigm called CloudDTH combining between digital twins and healthcare that is particularly dedicated to monitor elderly in their homes. The objective of CloudDTH is to improve medical services such that remote monitoring, diagnosing and predicting aspects of the health individual in terms of accuracy and speed.

## 1.8/ CONCLUSION

In this chapter, we presented a general overview about sensing-based technology that includes network architecture and types as well as the potential applications of sensor network. Then, we introduced the periodic data collection model along with the cluster-based scheme as an efficient and less-complex complex architecture for such networks. After that, we have described the challenges imposed in sensor networks while highlighting the energy consumption and the big data collection and management as the major challenges in such networks. Finally, we have presented a state-of-the-art to overcome the highlighted challenges while classifying them according to specific criteria.

# 2

# ON-IN: AN ON-NODE AND IN-NODE BASED MECHANISM FOR BIG DATA COLLECTION IN LARGE-SCALE SENSOR NETWORKS

## 2.1/ INTRODUCTION

The world has witnessed the bursting effects of sensor networks as a decisive element in any monitoring process whether in agriculture, medical care, environment or other fields. The large spread and usage of such networks is mainly due to three major reasons: their low-cost implementation, their flexibility, and their precision in yielding accurate data. Unfortunately, big data acquisition and transmission energy cost are two major problems that must be handled in order to maximize the lifetime of a network and its sensors. Therefore, data reduction techniques are becoming a fundamental operation to reduce the amount of transmitted data and consequently minimize the energy consumption.m

In this chapter, we assume that each node $N_i$ only contains one sensor $S_{i1}$ for the sake of simplicity. Thus, sensor node, node or sensor terms will refer to the same thing, and $N_i$=$S_{i1}$ as well as $M_i^p$=$R_{i1}^p$. Then, we propose a two phases data reduction mechanism dedicated to periodic large-scale sensor network applications: on-node and in-node. The final goal of our mechanism is to reduce data transmission, whether collected by the sensor nodes or transmitted by intermediate nodes, e.g. cluster-head (CH).

The rest of chapter is organized as follows. In Sections 2.2 and 2.3, we detail the on-node and the in-node models, respectively. Simulations and experiments are presented in Section 2.4. Section 3.6 concludes the chapter.

## 2.2/ SENSOR LEVEL: ON-NODE PREDICTION MODEL

In periodic sensor network, the huge amount of collected data and its corresponding huge number of transmitted packets lead to two sensor node problems: high level of energy consumption and sending unneeded/useless data to the sink. The first phase of our mechanism, e.g. on-node, is applied at the sensor node level and prevents sending similar data points sensed at each period $p$, based on a prediction model using the Newton's

forward difference method.

### 2.2.1/ NEWTON'S FORWARD DIFFERENCE METHOD

In numerical analysis, a Newton forward difference is an interpolation polynomial for a given set of data points. It estimates the value of a real function ($y_i = f(x_i)$) for any intermediate value of the independent variables ($x_i$).

**Definition 2.1** *Forward Differences. Given a set of $q+1$ data points, $\{(x_0, y_0), (x_1, y_1), \ldots, (x_j, y_j), \ldots, (x_q, y_q)\}$. The differences $\Delta y_0 = y_1 - y_0, \ldots, \Delta y_j = y_{j+1} - y_j, \ldots, \Delta y_{q-1} = y_q - y_{q-1}$ are called the first forward differences.*

Based on the above definition, we typically set up the forward difference table as:

| $x$ | $y$ | $\Delta y$ | $\Delta^2 y$ | $\ldots$ | $\Delta^{c-1} y$ | $\Delta^c y$ |
|---|---|---|---|---|---|---|
| $x_0$ | $y_0$ | | | | | |
| $x_1$ | $y_1$ | $\Delta y_0$ | $\Delta^2 y_0$ | $\ddots$ | | |
| $x_2$ | $y_2$ | $\Delta y_1$ | $\Delta^2 y_1$ | | $\Delta^{c-1} y_0$ | |
| $\vdots$ | $\vdots$ | $\Delta y_2$ | $\vdots$ | | | $\Delta^c y_0$ |
| $x_{q-2}$ | $y_{q-2}$ | | $\Delta^2 y_{q-3}$ | | $\Delta^{c-1} y_1$ | |
| $x_{q-1}$ | $y_{q-1}$ | $\Delta y_{q-2}$ | $\Delta^2 y_{q-2}$ | $\iddots$ | | |
| $x_q$ | $y_q$ | $\Delta y_{q-1}$ | | | | |

Then, in order to find the $y$-value corresponding to a new $x$-value ($x = x_0 + hu$), we use the Newton's Gregory forward interpolation formula:

$$y = f(x_0 + hu) = y_0 + u\Delta y_0 + \frac{u(u-1)}{2!}\Delta^2 y_0 + \cdots + \frac{u(u-1)(u-2)\ldots(u-c+1)}{c!}\Delta^c y_0 \quad (2.1)$$

This formula is particularly useful for interpolating the values of $f(x)$ near the beginning of the set of given values. $h$ is called the interval of difference ($h = x_1 - x_0$) and $u = (x - x_0)/h$, where $x$ is the value we want to find its corresponding $y$.

### 2.2.2/ ON-NODE PREDICTION ALGORITHM

The data collected by each sensor node, i.e. $R^p_{i1}$, are mainly redundant. Thus, in order to prevent sending redundant data to the CH, we propose to integrate the Newton's forward difference method into the sensor processing to reduce the data transmission to the CH. The idea is to find the coefficients of Newton Gregory equation then send them to the CH instead of sending the whole raw data in $R^p_{i1}$. Obviously, the data can be regenerated at any time based on the received equation.

The Newton Gregory polynomial needs $q+1$ data points to calculate the equation while the period contains $\mathcal{T}$ readings, where $\mathcal{T}$ is much bigger than $q+1$. Thus, we propose to select a subset of $d$ data points, named as $\mathcal{D}^p_{i1}$, from $R^p_{i1}$ to find the corresponding polynomial. $\mathcal{D}^p_{i1}$ can be formed based on the following equation:

$$\mathcal{D}^p_{i1} = \{(s_{1+t\times\lfloor\mathcal{T}/d\rfloor}, r^{i1}_{1+t\times\lfloor\mathcal{T}/d\rfloor}), (s_{\mathcal{T}}, r^{i1}_{\mathcal{T}})\} \quad (2.2)$$

where $r^{i1}_{1+t\times\lfloor\mathcal{T}/d\rfloor}$ are all readings collected at slot numbers $s_{1+t\times\lfloor\mathcal{T}/d\rfloor}$ (such that $t \in [0,\mathcal{T}]$ and $1+t\times\lfloor\mathcal{T}/d\rfloor < \mathcal{T}$) and $r^{i1}_{\mathcal{T}}$ is the last reading in $R^{p}_{i1}$.

After selecting the readings, the sensor computes the forward difference table in order to find the needed variables used in the Newton Gregory equation. Then, the sensor will send only the set of $R^{'p}_{i1} = \{x_0, x_1, y_0, \Delta y_0, \Delta^2 y_0, \ldots, \Delta^c y_0\}$ which is necessary to recalculate the $y$ values of all readings.

Finally, Algorithm 2.1 describes the on-node prediction model applied at each sensor node. Briefly, the algorithm takes the period size $\mathcal{T}$ as an input for the algorithm. After collecting data readings at each period (lines 1-5), the sensor node selects a set of readings, $\mathcal{D}^{p}_{i1}$, from $R^{p}_{ij}$ (lines 6-10). Finally, the sensor calculates the final set that will be sent to its CH based on the forward difference method and the Newton Gregory equation (lines 11-12).

**Algorithm 2.1** On-Node Prediction Algorithm.

**Require:** Sensor node number: $S_{i1}$, period size: $\mathcal{T}$, number of selected data points in NFD: $d$.
**Ensure:** Sent set: $R^{'p}_{i1}$.

1: $R^{'p}_{i1} \leftarrow \emptyset$
2: **for** $t = 1$ to $\mathcal{T}$ **do**
3: take reading value $r^{i1}_{t}$
4: $R^{p}_{i1} \leftarrow R^{p}_{i1} \cup \{r^{i1}_{t}\}$
5: **end for**
6: $\mathcal{D}^{p}_{i1} \leftarrow \emptyset$
7: **for** $t = 1$ to $\mathcal{T}/d$ **do**
8: $\mathcal{D}^{p}_{i1} \leftarrow \mathcal{D}^{p}_{i1} \cup \{(s_{(1+t\times\lfloor\mathcal{T}/d\rfloor)}, r^{i1}_{(1+t\times\lfloor\mathcal{T}/d\rfloor)})\}$
9: **end for**
10: $\mathcal{D}^{p}_{i1} \leftarrow \mathcal{D}^{p}_{i1} \cup \{r^{i1}_{\mathcal{T}}\}$
11: compute the forward difference table
12: find the variables of $R^{'p}_{i1}$

## 2.3/ CH Level: In-Node Clustering Model

At a periodic basis, the CH will receive all variable sets coming from all member nodes. Indeed, the spatial-temporal correlation among neighboring sensor nodes can produce a high redundancy among data sets that must be eliminated before sending final data to the sink. At the CH level, we propose to use a clustering approach in order to compress data coming from the sensors, so that only useful information are sent to the sink.

### 2.3.1/ Pattern-K-means Algorithm: PK-means

K-means has been considered as the most popular data clustering algorithms introduced in different domains. The idea behind K-means is to classify a number of datasets into $K$ clusters, where the similarity among datasets in the same cluster is high. The process of K-means starts by randomly selecting $K$ datasets as the centroids of the clusters, then

each dataset is assigned to the nearest centroid using a distance function. After that, the new centroids of the clusters are recalculated and the process is iterated until no more changes in the cluster centroids. Unfortunately, this traditional Kmeans suffers from the computation complexity due the distance calculation, especially when the number of datasets and the number of classes is high and each one contains a large number of values (like the WSN case). This leads to affect the data latency which is an important challenge in sensor networks, especially in critical applications.

In the literature, one can find many enhancements of K-means in order to overcome the data latency problem [144]. In this chapter, we propose a new version of K-means called Pattern K-means (PK-means) inspired from the work presented in [144]. PK-means can largely reduce the computation time of K-means and is suitable to sensor applications. After receiving the variable sets from all sensors, PK-means works based on the following steps:

- The CH regenerates the raw data, e.g. $R^p_{i1}$, for each sensor node based on the Newton's Gregory equation.
- For each regenerated dataset $R^p_{i1}$, PK-means calculates the following statistical parameters: $\mathcal{P}^p_{i1} = \{Peak, RMS, CrestFactor, Kurtosis, ImpulseFactor, ShapeFactor\}$. Consider that $\mathcal{P}_{i_j}$ refers to any element in $\mathcal{P}^p_{i1}$ such as $\mathcal{P}_{i_0}$ corresponds to $Peak$, $\mathcal{P}_{i_1}$ corresponds to $RMS$ and so on. The parameters used in our pattern can be calculated as follows:

$$Peak = \frac{1}{2}\left(max(r^{i1}_t) - min(r^{i1}_t)\right) \quad RMS = \sqrt{\frac{1}{\mathcal{T}}\sum_{t=1}^{\mathcal{T}}(r^{i1}_t - \overline{r^{i1}_t})^2}$$

$$CrestFactor = \frac{Peak}{RMS} \qquad Kurtosis = \frac{\frac{1}{\mathcal{T}}\sum_{t=1}^{\mathcal{T}}(r^{i1}_t - \overline{r^{i1}_t})^4}{RMS^4}$$

$$ImpulseFactor = \frac{Peak}{\frac{1}{\mathcal{T}}\sum_{t=1}^{\mathcal{T}}|r^{i1}_t|} \qquad ShapeFactor = \frac{RMS}{\frac{1}{\mathcal{T}}\sum_{t=1}^{\mathcal{T}}|r^{i1}_t|}$$

- PK-means selects randomly $K$ sets among $\mathcal{P}^p_{i1}$ sets as the initial cluster centroids.
- To assign a dataset to a cluster, PK-means calculates the Manhattan distance between $\mathcal{P}^p_{i1}$ and all the cluster centroids. Subsequently, given two statistical parameters $\mathcal{P}^p_{i1}$ and $\mathcal{P}^p_{j1}$, the Manhattan distance can be calculated according to the following equation:

$$D_{MH}(\mathcal{P}^p_{i1}, \mathcal{P}^p_{j1}) = \sum_{k=1}^{6} |\mathcal{P}_{i_k} - \mathcal{P}_{j_k}| \tag{2.3}$$

- Like K-means, the process continues until no more changes in the cluster centroids.

Algorithm 2.2 shows how PK-means is working out. First, the CH calculates the parameters of $\mathcal{P}^p_{i1}$ for each set $R^p_{i1}$ sent by the sensor. Then, it randomly selects $K$ centroids as the initial centers of the clusters. After that, the Manhattan distance is calculated between every sensor pattern and the cluster centroids while the data sensor is assigned to the nearest one. A loop is done until no change in the cluster centroids. Finally, the nearest data set to the center in order to send to the sink as a representing of the cluster.

**Algorithm 2.2** PK-means Algorithm.

**Require:** Sensor datasets: $R^p = \{R^p_{11}, R^p_{21}, \ldots, R^p_{n1}\}$, number of clusters: $K$.
**Ensure:** Set of clusters: $C = \{C_0, C_1, \ldots, C_{K-1}\}$.
1: **for** each set $R^p_{i1} \in R^p$ **do**
2: // calculate the parameters of pattern $\mathcal{P}^p_{i1} = \{Peak, RMS, CrestFactor, Kurtosis, ImpulseFactor, ShapeFactor\}$
3: **for** each parameter $i_j \in \mathcal{P}^p_{i1}$ **do**
4: calculate $\mathcal{P}_{i_j}$
5: **end for**
6: **end for**
7: randomly choose $K$ centroids $G_i$ ($i \in [0, \ldots, K-1]$) for the clusters
8: $D_{MH} = 0$
9: **repeat**
10: **for** each set $\mathcal{P}^p_{i1} \in \mathcal{P}^p$ **do**
11: calculate $D_{MH}(\mathcal{P}^p_{i1}, G_j)$ where $j \in [0, \ldots, K]$
12: consider $D_{MH}(\mathcal{P}^p_{i1}, G_m) < D_{MH}(\mathcal{P}^p_{i1}, G_{m^*})\ \forall\ m^* \in [0, \ldots, K] - [m]$
13: Assign $\mathcal{P}^p_{i1}$ to the cluster $C_m$
14: **end for**
15: Update the centroid $G_m$ of each cluster $C_m$
16: **until** clusters' centroids no longer changes
17: **return** $C$

## 2.4/ Performance Evaluation

In order to evaluate the performance of our mechanism, both simulations and real experiments have been conducted.

### 2.4.1/ Simulation Results

In our simulations, we used the scalar dataset picked up from sensors deployed in the Intel Berkeley Research lab [145] (Figure 2.1). This data contains readings for $46$ sensors recording environmental condition including temperature, humidity, light and voltage. Every $31$ seconds, the sensor collects new reading for each feature then it sends toward the sink for archive purpose. For the sake of simplicity, we only considered the temperature readings in each node where we used a file that includes a log of about $50000$ readings for temperature readings. We assume that each sensor node reads the data from its corresponding file for a period of time, then it sends them toward a CH placed at the center of the lab after applying our mechanism. We implemented our technique based on Java simulator and we compare the results to those obtained with prefix frequency filtering (PFF) [61].

Table 2.1 summarizes the parameters used in our simulation with their tested values.

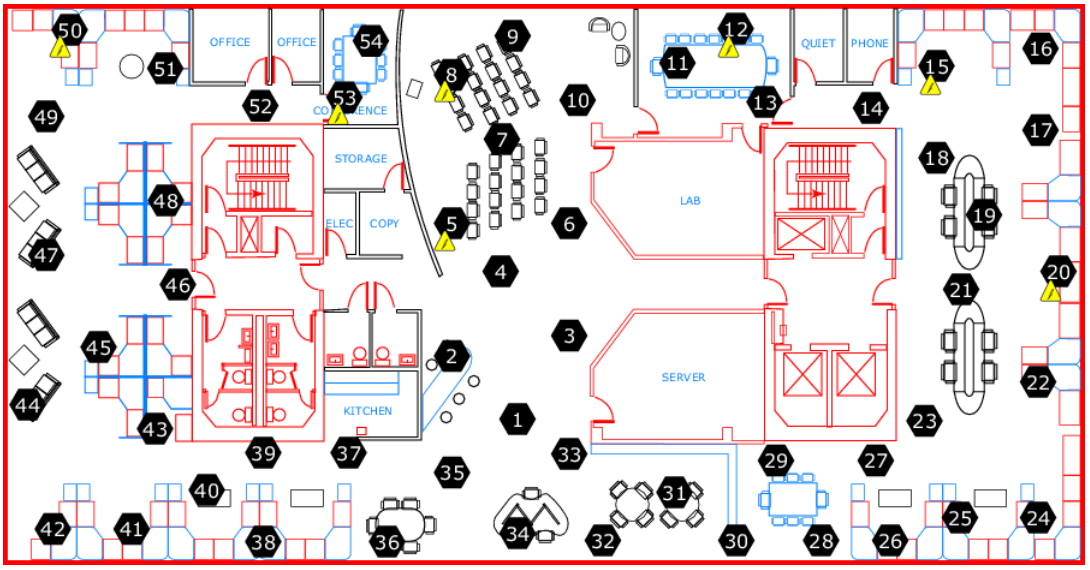



Figure 2.1: Distribution map of the sensors in the Intel lab.

| Parameters | Values |
|---|---|
| Dimension of area | $42 \times 33$ m$^2$ |
| Number of sensors | 46 |
| Measured physical parameter | temperature |
| Number of readings | 2.3 millions |
| Slot interval | 31 seconds |
| Period size ($\mathcal{T}$) | 50, 100, 250, 500 |
| number of selected data points in NFD ($d$) | 4, 5, 6, 7, 8 |
| Number of clusters ($K$) | 5, 6, 7, 8 |

Table 2.1: Simulation parameters.

#### 2.4.1.1/ Raw Data vs Recovered Data

Figure 2.2 shows the performance of an on-node phase by recovering raw data collected by the sensor nodes after applying the Newton Gregory equation (referred as NG in the figure). We fixed the period size $\mathcal{T}$ to $100$ readings and we varied the number of selected points ($d$) to $4$, $6$ and $8$. The results show that the on-node phase gives a high data accuracy level compared to the raw data. We can also notice that, the accuracy of the recovered data increases by increasing the number of selected points $d$. This is because, the accuracy of Newton Gregory formula increases when $d$ increases.

#### 2.4.1.2/ Sensor Data Transmission Ratio

This section studies the average number of readings sent from each sensor to the CH (Figure 2.3). Compared to PFF technique that uses data aggregation approach, that figure shows that on-node phase gives better results in terms of eliminating redundancy and reducing data transmission to the CH. Subsequently, it reduces the amount of data transmission from $20\%$ to $84\%$ compared to PFF when varying $\mathcal{T}$ (Figure 2.3(a)), and from

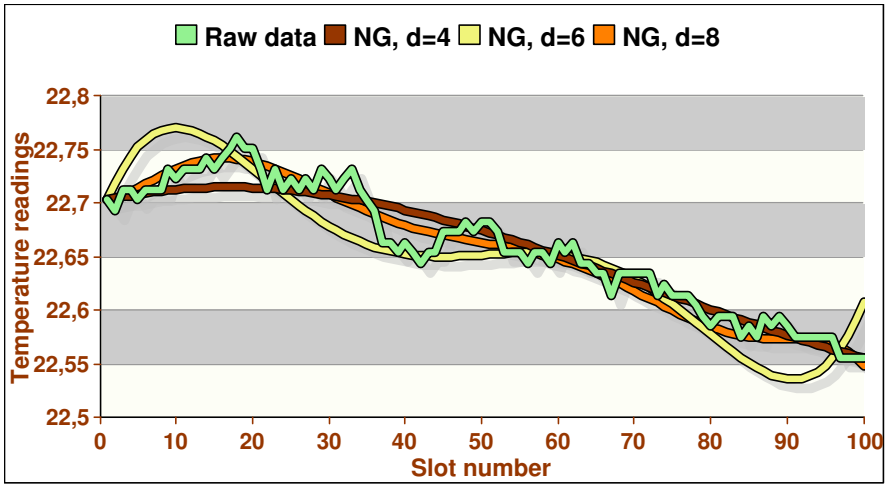
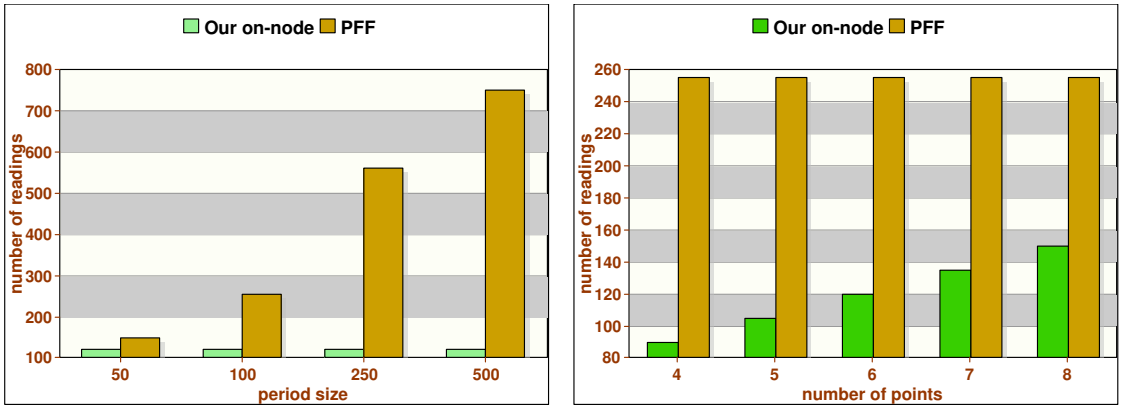



Figure 2.2: Comparison between raw and Newton Gregory generated data, $\mathcal{T} = 100$

$41\%$ to $64\%$ when varying $d$ (Figure 2.3(b)). This reduction is because, the sensor node only sends, using on-node phase, the Newton Gregory coefficients to the CH; while in the PFF, it uses an aggregation method to send a portion of collected data instead of the whole raw data. Therefore, on-node phase will highly minimize the energy consumption in the sensor and increase its lifetime.



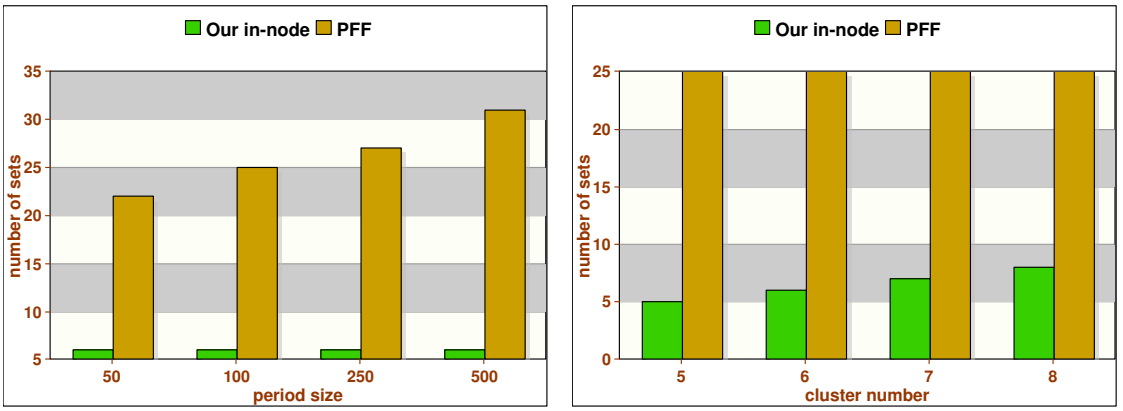

(a) $d = 6$ (b) $\mathcal{T} = 100$

Figure 2.3: Number of readings periodically sent to the CH.

### 2.4.1.3/ CH Data Transmission Ratio

Figure 2.4 shows the CH data transmission ratio or the periodic number of sets sent to the sink after applying our in-node phase and PFF. Figure 2.4(a) shows the effects of varying the period size $\mathcal{T}$ while Figure 2.4(b) presents the effects of varying the number of clusters $K$ (from $5$ to $8$). The obtained results show that in-node phase can successfully reduce the data transmission ratio at the CH, compared to PFF. We notice that in-node phase

reduces up to $80\%$ when varying $F$ and $K$. This confirms the fact that data clustering is an efficient approach to find redundancy among datasets. Therefore, our mechanism can be considered as an energy-efficient technique for both sensor and CH levels.




(a) $K = 6$ (b) $\mathcal{T} = 100$

Figure 2.4: Periodic number of sets sent to the sink, $d = 6$.

### 2.4.2/ Experiment Results

This section shows the results of real data experiments made in our laboratory with dimension $22 \times 12\, m^2$. We deployed twenty telosB motes in order to collect temperature and humidity data where data are sent to a sink of type SG1000 [146], which it is connected to a laptop machine ($16$ GB RAM with $8$ CPUs of $2.7$ GHz) in order to retrieve and make statistics over the collected data. TelosB uses TinyOS and can be programmed based on nesC language [147]. The sampling rate of all the sensors has been set to $1$ reading per $30$ seconds while the period size is set to $50$ readings. Nodes positions in our laboratory are shown in Figure 2.5 with identifiers (IDs) ranging from $1$ to $20$ as well as an ID $= 0$ is assigned to the SG1000.

#### 2.4.2.1/ Raw Data vs Recovered Data

In this section, our objective is to show the relevance of on-node phase comparing among simulations and experiments. Similar to Figure 2.2, Figure 2.6 shows the difference between raw recovered data after applying on-node phase for both temperature and humidity sensors. As expected, the on-node algorithm allows to save a high accuracy level of recovered data. This can be noticed through the nearest distance between raw and recovered data at both sensors. Compared to the simulation results (Figure 2.2), the experimentations conducted in our lab confirms the behavior of our on-node phase concerning the reducing of data transmission ratio while conserving a high level of information integrity.

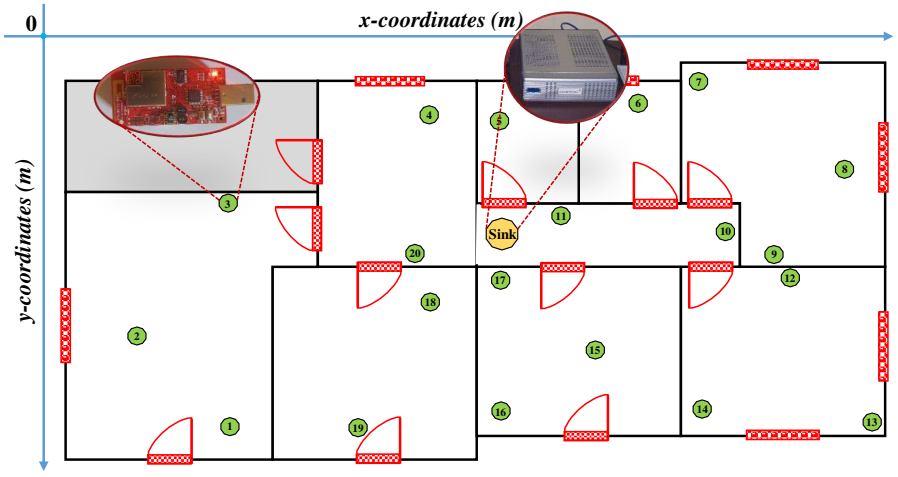



Figure 2.5: Distribution of nodes in our lab.

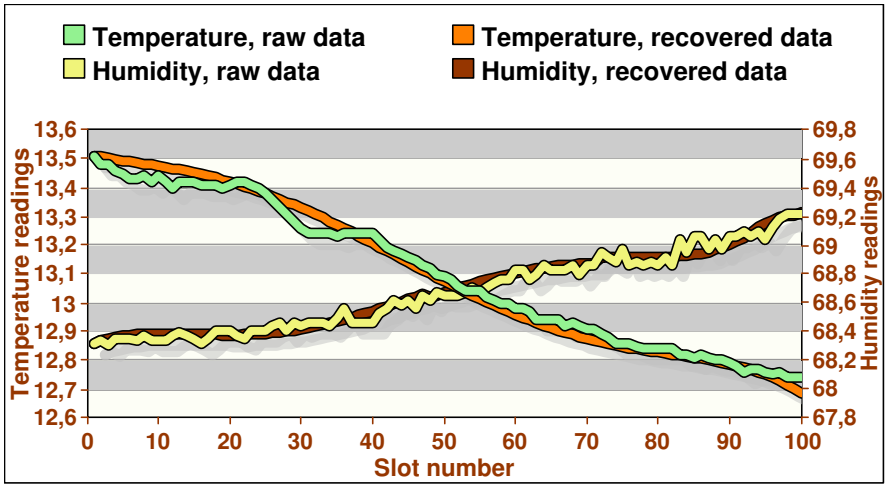



Figure 2.6: Comparison between raw and recovered data, $d = 6, \mathcal{T} = 100$.

### 2.4.2.2/ Iteration Loop Number

Figure 2.7 shows the number of iterations needed by PK-means algorithm in order to find the final clusters. Obviously, more the number of iterations in PK-means increases more the packet delivery time to the sink becomes. Thus, data latency will be highly affected. The results show that PK-means needs approximately $4$ iteration loops to converge, in both temperature and humidity. This value is likely acceptable compared to that needed by the traditional K-means. Therefore, PK-means can be considered as an efficient data latency algorithm that seems very suitable to the sensor network case.

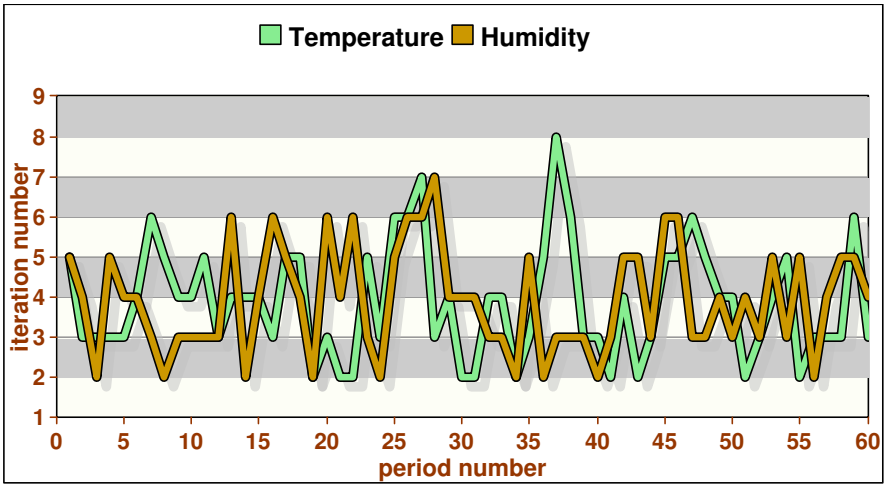



Figure 2.7: Number of iterations in applying PK-means, $\mathcal{T} = 100, d = 6, K = 3$.

## 2.5/ CONCLUSION

With a constant rise in the importance of sensor networks in multiple fields, the need for development of new big data reduction mechanisms is becoming essential. In this chapter, we proposed an on-node and in-node (ON-IN) mechanism for reducing big data collected in sensor networks. The first phase of our technique focuses on reducing the data transmitted by sensors using the Newton's forward difference method. The second phase focuses on reducing the data generated by neighboring nodes using PK-means algorithm. The proposed mechanism is evaluated using both simulations and experiments on telosb motes. Our results demonstrated that the proposed mechanism is better than other techniques in terms of data transmission and energy consumption.

# 3

# Adaptive Strategy and Decision-Making Model for Sensing-Based Network Applications

## 3.1/ Introduction

Nowadays, we need to collect and store huge amount of information about surroundings and people behaviors. Information surroundings are mostly used by governments in order to monitor natural phenomena and thus predicting any possible disaster. Otherwise, information about people activities on research or social media, etc., are used by business in order to understand, analyze and make target advertisements. Such need of monitoring makes sensor network applications one of the most active research field today. Generally, a sensor network consists of a spatially distributed autonomous sensor nodes that aims at monitoring physical or environmental conditions and to cooperatively pass their data through the network to a sink node.

In sensor network, energy conservation and decision-making attract a great attention in the literature. In one hand, the sensor nodes have limited energy power, which is mostly not rechargeable especially in hostile environments, and the data transmission is highly cost operation in WSN. On the other hand, the sensor nodes are usually densely deployed in order to monitor the interest area which results in a huge amount of data collected. These characteristics make the energy conservation and the decision-making a major challenge for sensor networks. Therefore, data redundancy reduction becomes a necessary operation for sensor networks thus the energy-saving is raised and the decision-making is enhanced.

In this chapter, we are interested in removing redundancy starting by the raw data collected at the sensor nodes and arriving to making decisions at the sink. First, we assume a cluster-based architecture for the sensor network, where a CH is assigned for each cluster. Similar to chapter $2$, we also assume that each node $N_i$ only contains one sensor $S_{i1}$ for the sake of simplicity. Then, we propose filtering methods for each tier of the network (sensor nodes, CH and sink). Finally, we evaluate our technique in terms of energy conserving and information integrity through simulations on real sensor data.

The rest of this chapter is organized as follows: Sections 3.2, 3.3 and 3.4 present our

methods applied on the sensors, CH and sink respectively. The simulation results are presented in Section 4.6. The conclusion is reported in section 3.6.

## 3.2/ SENSOR TIER: DIVIDE-AND-CONQUER ALGORITHM

In this chapter, we are interested in periodic data collection model in which each sensor node $S_{i1} \in N_i$ collects a set of $\mathcal{T}$ readings, e.g. $R_{i1}^p = \{r_1^{i1}, r_2^{i1}, \ldots, r_{\mathcal{T}}^{i1}\}$, during each period $p$ before sending to the CH.

### 3.2.1/ DIVIDE-AND-CONQUER ALGORITHM

Mostly, the dynamic of the monitored condition, that slow down and speed up, produces a huge redundancy among the collected data by each sensor, especially when the slot time is short. Our objective at this part is to eliminate redundant readings among $R_{i1}^p$ and send a useful information to the CH. Our idea is to divide the readings in $R_{i1}^p$ into $\mathcal{V}$ equal divisions then, each division is represented by only one information, e.g. the mean value. This allows to reduce the size of $R_{i1}^p$ while the accuracy of the integrity of the information will be preserved (since successive readings in a division are mostly redundant).

Algorithm 3.1 describes the divide-and-conquer algorithm which is periodically applied over the data collected by each sensor. The process starts by dividing the reading vector $R_{i1}^p$ into $\mathcal{V}$ equal divisions, where each division contains $\mathcal{T}/\mathcal{V}$ readings (lines 2-3). Then, the mean value for each division is calculated and inserted to the final vector that will be sent to the CH (lines 4-10).

---

**Algorithm 3.1** Divide-and-Conquer Algorithm.

---

**Require:** Reading vector: $R_{i1}^p = \{r_1^{i1}, r_2^{i1}, \ldots, r_{\mathcal{T}}^{i1}\}$, period size: $\mathcal{T}$, division number: $\mathcal{V}$.
**Ensure:** Mean reading vector of $R_{i1}^p$: $M_{i1}^p$.

1: $M_{i1}^p \leftarrow \emptyset$
2: $rpd = \mathcal{T}/\mathcal{V}$ // $rpd$ is a temporary variable
3: **for** each division $\mathcal{V}_j \in R_{i1}^p$ **do**
4: $sum = 0$
5: **for** each reading $r_j^{i1} \in \mathcal{V}_j$ **do**
6: $sum = sum + r_j^{i1}$
7: **end for**
8: $M_{i1}^p = M_{i1}^p \cup \{sum/rpd\}$
9: **end for**
10: return $M_{i1}^p$

---

### 3.2.2/ ILLUSTRATIVE EXAMPLE

In Figure 3.1, we show an illustrative example for the process of divide-and-conquer described in Algorithm 3.1. Given a period of 15 readings with a number of divisions $\mathcal{V}$

equals to 3, thus each division contains 5 readings. Then, the sensor node will send a set of 3 mean values to the CH at the end of the period.

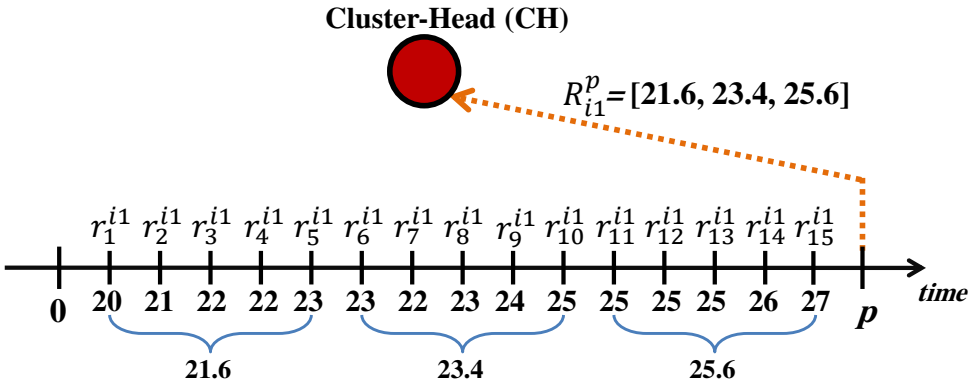



Figure 3.1: Illustrative example for divide-and-conquer algorithm.

## 3.3/ CH TIER: SUPPORT-CONFIDENCE METHOD

At the end of each period, the CH receives the mean sets coming from its sensor cluster. Our objective at the second tier is to allow CH to reduce the redundancy from data collected by neighboring nodes at the same cluster. Then, the CH sends a useful information representing the status of the monitored condition at each cluster to the sink. This is done using the support-confidence algorithm proposed in the next section.

### 3.3.1/ SUPPORT-CONFIDENCE ALGORITHM

As mentioned before, the sensor nodes belong to the same cluster have a high spatial-temporal correlation, especially when the cluster dimension gets smaller. Like Apriori and association rules algorithms [148], our algorithm searches for the frequent items in the received mean sets then, it sends them to the sink. In order to increase the accuracy of the information, the frequent items are selected according to a defined confidence threshold $C$. Our Support-Confidence algorithm is based on the following definitions.

**Definition 3.1** *Frequent Mean. A mean value $m_t$ is defined as a frequent mean if its support is greater or equal than a confidence threshold $C$ as follows:*

$$Sup(m_t) \geq C, \text{ where } C \in ]0, \infty[ \tag{3.1}$$

Algorithm 3.2 shows the support-confidence process applied at the CH nodes. For each received mean value, the CH searches for equal values. If an equal mean is found, it increments its support by that of found mean; else, the mean is added to the temporary list with a support equals to 1 (lines 1-15). Then, the CH only sends the means with support greater or equal to the confidence threshold (lines 16-22).

**Algorithm 3.2** Support-Confidence Algorithm.

**Require:** Mean reading vectors: $M^p = [M^p_{11}, M^p_{21}, \ldots, M^p_{n1}]$, confidence threshold: $C$.
**Ensure:** Frequent mean set: $F^p = \{(m_1, Sup(m_1)), (m_2, Sup(m_2)), \ldots, (m_k, Sup(m_k))\}$.

```
1: T_p ← ∅ // create a temporary list
2: for each mean vector M^p_{i1} ∈ M^p do
3:    for each mean reading value m_t ∈ M^p_{i1} do
4:       if T_p is empty then
5:          Sup(m_t) ← 1
6:          T_p ← T_p ∪ {(m_t, Sup(m_t))}
7:       else
8:          for each pair (m_k, Sup(m_k)) ∈ T_p do
9:             if m_t = m_k then
10:               Sup(m_t) = Sup(m_t) + Sup(m_k)
11:            end if
12:         end for
13:      end if
14:   end for
15: end for
16: F^p ← ∅
17: for each pair (m_t, Sup(m_t)) ∈ T_p do
18:   if Sup(m_t) ≥ C then
19:      F^p ← F^p ∪ {(m_t, Sup(m_t))}
20:   end if
21: end for
22: return F^p
```

### 3.3.2/ Illustrative Example

Figure 3.2 shows an illustrative example for the support-confidence algorithm where 3 sensor nodes with their CH have been deployed. After receiving mean sets coming from all sensors, the CH searches the support value for all means. Then, it only selects means whose supports are greater than 3 (i.e. $C = 3$) in order to send them toward the sink, i.e. $[(20, 5), (22, 4), (23, 3)]$.

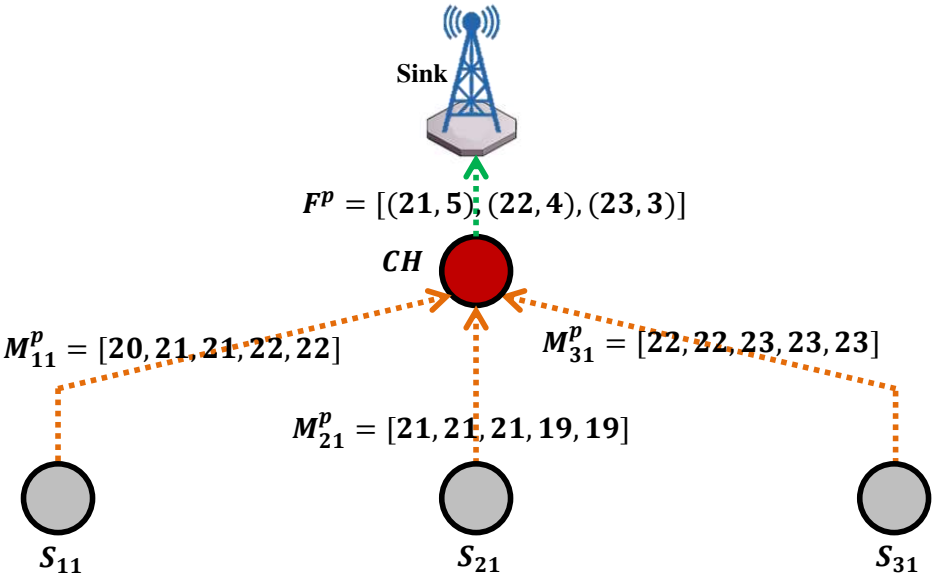



Figure 3.2: Illustrative example for support-confidence algorithm.

## 3.4/ Sink Tier: Decision-Making Model

Decision-making is the main target behind deploying sensor networks. Unfortunately, most of the proposed techniques have focused on overcoming challenges exposed by sensor networks like network lifetime, sensor localization, security, etc. Whilst, few researchers were interested to propose decision-making techniques at the sink nodes. In the last stage of our mechanism, we build a model that allows decision makers to take real-time decisions about the monitored condition. One of the strong advantages of our model that it is not dedicated to a specific sensor network application and it can be customizable depending on the application requirements.

### 3.4.1/ Real-Time Decision Model

In this section, we describe our cross-applications decision model which is based on two main tables: score decision table and early decision table. The score decision table is a customizable guide used by the application services staff in order to determine the real-time status of the monitored zone. According to the decision makers, a normal range, $]r_l^j, r_u^j[$, is defined for each physical condition $j$ monitored by the sensor $S_{ij}$ in $N_i$. Readings outside of this range are assigned a weighted score indicating the criticality degree of the collected readings; more the reading is deviated from the range, more the criticality degree is. For the sake of simplicity, we represent the criticality of the condition situation by a score ranging in $[0,3]$ where $0$ indicates a normal situation and $3$ indicates a severe condition. Figure 3.3 shows the customized score table for all conditions monitored by the sensors in a node. After determining the lower $(r_l^j)$ and upper $(r_u^j)$ bounds of the normal range of each condition $j$, a threshold $\delta_j$ is defined in order to determine the deviation of the readings from the normal range. Therefore, a set of thresholds is defined for all sensors in a node as follows: $\mathcal{H} = \{\delta_1, \delta_2, \ldots, \delta_Q\}$; $\mathcal{H}$ is a user-defined set of thresholds determined according to the application requirements.

| Score | 3 | 2 | 1 | 0 | 1 | 2 | 3 |
|---|---|---|---|---|---|---|---|
| Condition 1 | $\leq r_l^1 - 2\delta_1$ | $]r_l^1 - 2\delta_1, r_u^1 - 2\delta_1[$ | $]r_l^1 - \delta_1, r_u^1 - \delta_1[$ | $]r_l^1, r_u^1[$ | $]r_l^1 + \delta_1, r_u^1 + \delta_1[$ | $]r_l^1 + 2\delta_1, r_u^1 + 2\delta_1[$ | $\geq r_u^1 + 2\delta_1$ |
| Condition 2 | $\leq r_l^2 - 2\delta_2$ | $]r_l^2 - 2\delta_2, r_u^2 - 2\delta_2[$ | $]r_l^2 - \delta_2, r_u^2 - \delta_2[$ | $]r_l^2, r_u^2[$ | $]r_l^2 + \delta_2, r_u^2 + \delta_2[$ | $]r_l^2 + 2\delta_2, r_u^2 + 2\delta_2[$ | $\geq r_u^2 + 2\delta_2$ |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| Condition Q | $\leq r_l^Q - 2\delta_Q$ | $]r_l^Q - 2\delta_Q, r_u^Q - 2\delta_Q[$ | $]r_l^Q - \delta_Q, r_u^Q - \delta_Q[$ | $]r_l^Q, r_u^Q[$ | $]r_l^Q + \delta_Q, r_u^Q + \delta_Q[$ | $]r_l^Q + 2\delta_Q, r_u^Q + 2\delta_Q[$ | $\geq r_u^Q + 2Q$ |

Figure 3.3: Customizable score table.

After creating the score table, we define the early decision table (EDT) to take an appropriate decision for each cluster zone. Figure 3.4 shows the customizable EDT where an action is predetermined by the users when the aggregated score for collected data matches a predefined range of scores. Indeed, the score range should be determined according to the criticality of the monitored application. Therefore, when the sink receives the mean set coming from a CH, it searches the score for each mean value from the score table. Then, it calculates the aggregated (total) score for the whole set. Consequently, a real-time decision is taken based on the aggregated score.

| Aggregated Score | 0 | $[S_1, S_2]$ | $]S_2, S_3]$ or having one score = 3 | $> S_3$ |
|---|---|---|---|---|
| Description | Normal changing | A bit changing is noticed | Rapid changing is detected | Physical parameter is very dynamic |
| Action | No action is needed | Take action 1 (be ready) | Take action 2 (almost critical) | Take action 3 (critical status) |

Figure 3.4: Early decision table.

## 3.5/ Simulation Results

This section shows the results of our technique comparison with the prefix frequency filtering (PFF) technique proposed in [61]. We used the real temperature data collected from the $46$ sensors deployed in the Intel Berkeley lab [145]. We divided the network into two equal clusters which have $CH_1$ and $CH_2$ as clsuter-heads respectively (Figure 3.5). Thus, the sensors send their data periodically to their appropriate cluster-heads. In our simulation, we varied the parameter values as follows:

- The period size ($\mathcal{T}$) is set to various values such as: $50$, $100$ and $200$ readings.
- The number of divisions ($\mathcal{V}$) is set to various values such as: $5$ or $10$.

### 3.5.1/ Data Reduction Ratio at Sensors

In this section, we study the number of data values periodically sent from each sensor to the CH (Figure 3.6). Indeed, the number of remaining data in our technique is highly

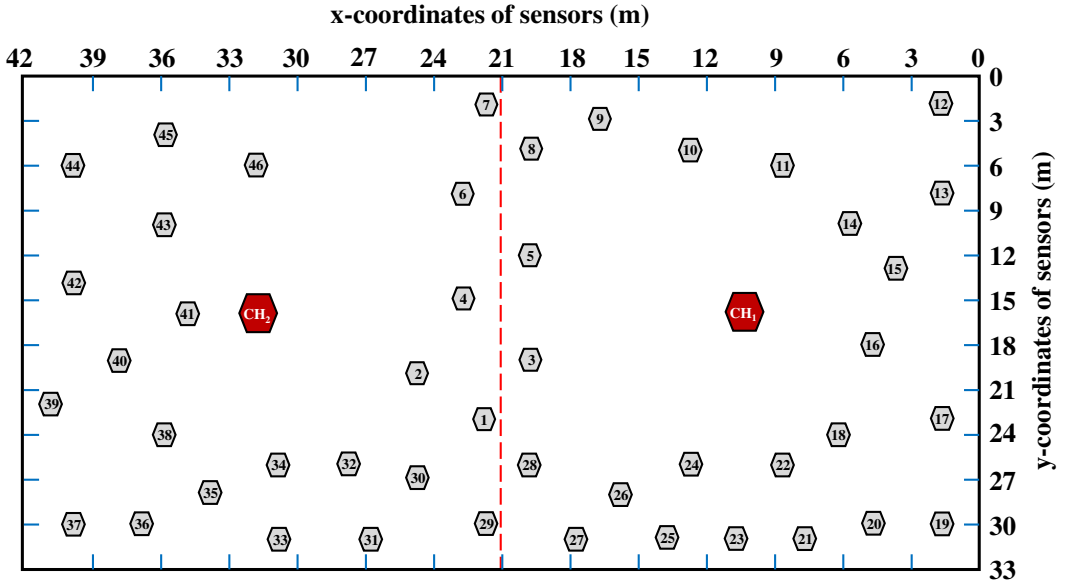
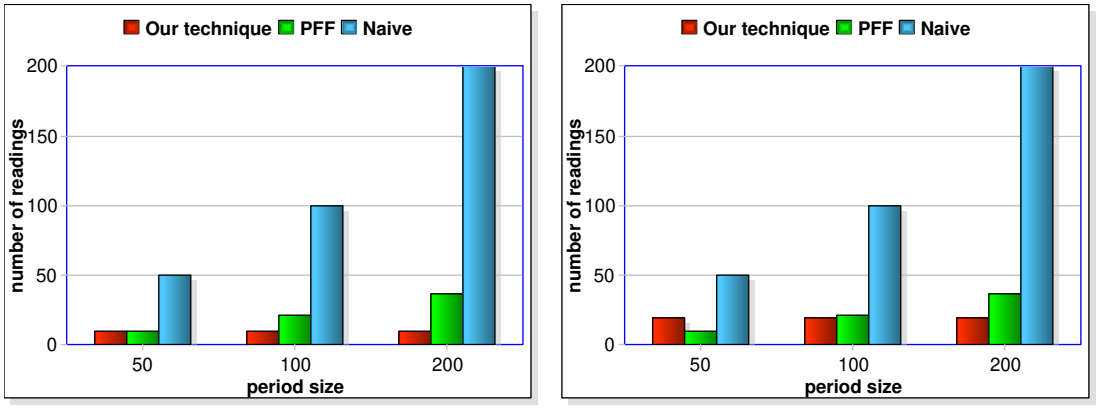



Figure 3.5: Distribution of sensor nodes and CHs in the Intel Lab.

dependent on the number of divisions while it depends on the local aggregation phase used in PFF. The results of Figure 3.6 show that the divide-and-conquer algorithm used in our technique allows sensors to more eliminate redundancy and reducing data transmission to the CH, compared to PFF method. Subsequently, in best cases, it reduces up to $70\%$ compared to PFF when fixing $\mathcal{V}$ to $5$ (Figure 3.6(a)), and up to $45\%$ when fixing $\mathcal{V}$ to $10$ (Figure 3.6(b)). Indeed, the sensor node searches, using divide-and-conquer algorithm, the similarity between successive readings in a period while in the PFF, it uses an aggregation method to send a portion of collected data instead of the whole raw data. Therefore, our technique can be efficiently used at sensors side, minimizing their energies and increasing their lifetimes. We can also notice that both techniques send more data to the CHs when the period size ($\mathcal{T}$) increases.

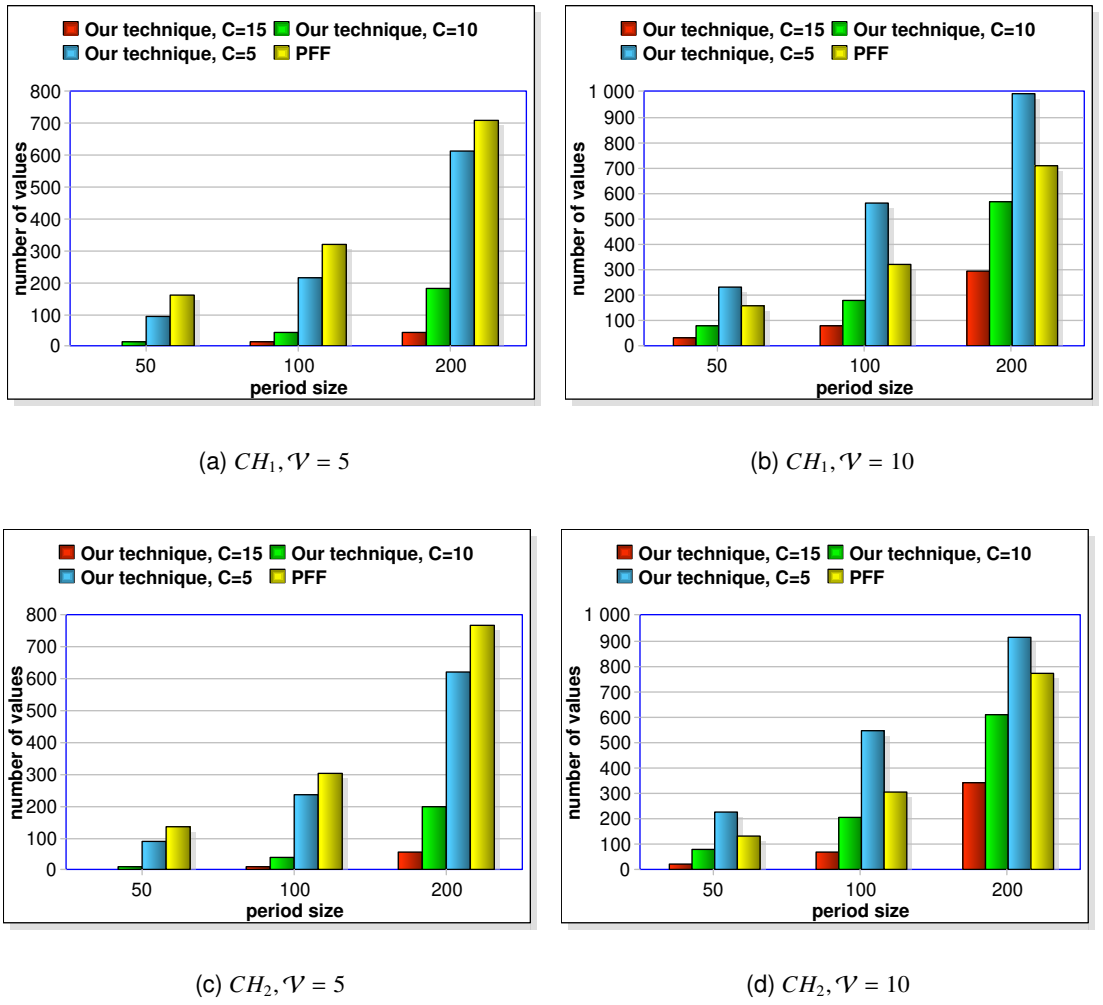



(a) $\mathcal{V} = 5$ (b) $\mathcal{V} = 10$

Figure 3.6: Number of periodic readings sent from each sensor to the CH.

### 3.5.2/ Data Reduction Ratio at CHs

In this section, we study the size of final data periodically sent from each CH to the sink node, after applying the support-confidence algorithm proposed in our technique and the in-network aggregation used in PFF. The obtained results of our technique are highly dependent on the confidence threshold $C$ (defined in Algorithm 3.2), where its value can be selected based on the monitored application and the number of sensors in each cluster. Figure 3.7 (a and b) and Figure 3.7 (c and d) show the obtained results for cluster-heads $CH_1$ and $CH_2$ respectively, when changing $\mathcal{T}$, $C$ and $\mathcal{V}$. We observe that our technique outperforms PFF technique in terms of reducing data transmission, except when $\mathcal{V}$ and $C$ have high values. Apart from exception cases, the support-confidence algorithm reduces up to $97\%$ of data transmitted to the sink for both $CH_1$ and $CH_2$.




(a) $CH_1, \mathcal{V} = 5$

(b) $CH_1, \mathcal{V} = 10$

(c) $CH_2, \mathcal{V} = 5$

(d) $CH_2, \mathcal{V} = 10$

Figure 3.7: Size of periodic data sent from each CH to the sink.

Based on the results of Figure 3.7, the following observations can be also noticed:

- The cluster-heads send more data to the sink when the confidence threshold $C$ decreases. This is because the constraint of criticality will be more flexible and thus, more data will meet the value of $C$.

- The size of data transmitted to the sink will increase when the division number $\mathcal{V}$ increases. This is because, each CH will receive more data from the sensors (see results of Figure 3.6).
- The data transmission ratio to the sink will increase with increasing of the period size $\mathcal{T}$. This is because, the dissimilarity between the collected data will increase which makes more readings meeting the threshold $C$ at the CHs.
- The results at both cluster-heads are almost similar. This is done due to the similar distribution of the sensor nodes in both clusters.

### 3.5.3/ Decision Results at the Sink

After receiving datasets sent from both cluster-heads, the sink uses score and decision tables in order to take the appropriate decision for each cluster. As mentioned before, the customization of both tables are highly related to decision makers and application staff that carefully define the values of range threshold in both tables. Figures 3.8 and 3.9 show the customizable score and decision tables proposed for the temperature condition monitored in the Intel Lab. In the score table, we fixed the value of $\delta_j$ to $1$ Celsius degree in order to determine the deviation from the normal range (as chosen in $]18, 19[$). On the other hand, a set of actions have been proposed in order to inform people inside the lab about what they have to do when the temperature varies.

| Score | 3 | 2 | 1 | 0 | 1 | 2 | 3 |
|---|---|---|---|---|---|---|---|
| Temperature | $\leq 16$ | $]16, 17]$ | $]17, 18]$ | $]18, 19[$ | $[19, 20[$ | $[20, 21[$ | $\geq 21$ |

Figure 3.8: Score table customized to temperature monitoring.

| Aggregated Score | 0 | $[1, 10]$ | $]10, 30]$ or having one score = 3 | $> 30$ |
|---|---|---|---|---|
| Decision number | 0 | 1 | 2 | 3 |
| Action | Continue routine in the lab | Alert: People needs water once every 2-3 hours Or keep warm by puting more layers | Attention: Body needs water once every 1 hour Or a coat is essentiel to keep warm | Danger: High temperature can lead to many illnesses (Leave the lab) |

Figure 3.9: Early decision table customized to temperature monitoring.

Figure 3.10 shows the real-time decision made by the sink according to the periodic data received from each cluster-head ($CH_1$ and $CH_2$). In the simulation, we monitored the temperature condition for $15$ periods, where we fixed the period size to $100$ readings, the division number to $5$ and the confidence threshold to $10$. The results show two observations: first, the temperature condition is differently changed in each zone at the lab during periods. Second, data sent from the $CH_1$ are more critical compared to those sent

from the $CH_2$. Therefore, our technique is considered as a real-time helpful methods for decision makers.

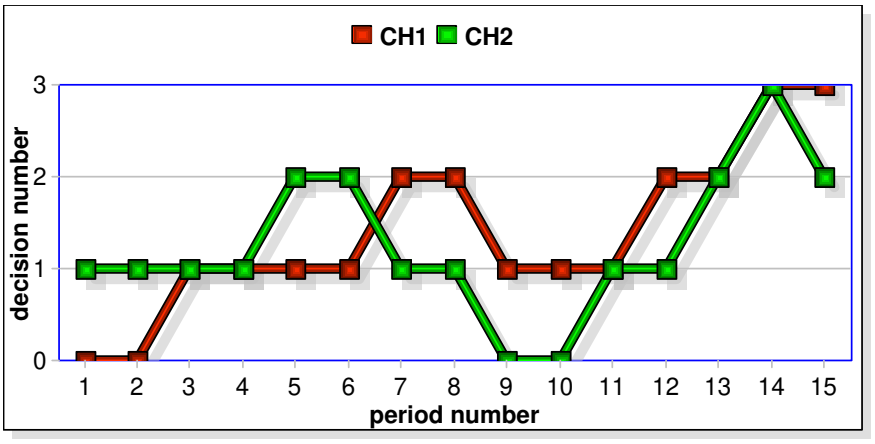



Figure 3.10: Variation of decision-making at the sink during periods, $\mathcal{T} = 100, \mathcal{V} = 5, C = 10$.

## 3.6/ CONCLUSION

Data redundancy reduction is an essential operation in sensor network that allows to remove unneeded and useless data sent to the sink thus, saves network energy and helps in taking decisions. In this chapter, we have proposed an energy-efficient adaptive strategy and decision-making technique dedicated to periodic sensor applications. Our technique used cluster-based network architecture where every tier of the network (sensor, CH and sink) applied a specific filtering model. The sensor tier used a divide-and-conquer algorithm; the CH tier applied a support-confidence method while the sink tier used two customizable tables (score table and decision table) to take real time decision about each cluster on the network. Through simulation on real sensor data, we have demonstrated the efficiency of our technique in terms of energy saving and decision-making, while conserving the accuracy of the information.

# 4

# All-in-One: Toward Hybrid Data Collection and Energy Saving Mechanism in Sensing-Based Applications

## 4.1/ Introduction

The applications of sensor networks are numerous and promising, and they are expanding into new applications every day. Indeed, sensor networks are by nature heterogeneous and dense. From one hand, the heterogeneity is represented by the various types of sensors, e.g. scalar and multimedia, that need not only to coexist but also to be managed with common operational objectives. On the other hand, the high density means having large number of sensor nodes covering a given area, and generating large volumes of data that, in many cases, contain a large amount of redundant information. Such redundancy might be due to the close proximity of the sensor nodes and the overlap in a given area of coverage, or to the fact that the sampling rate of data is faster than the speed of the variation of the monitored variables. Unfortunately, a consequence of such redundancy is an excess of energy consumption during data collection, processing, and transmission. In its turn, such increasing of energy consumption leads to shortening the operational lifetime of the networks and decreasing the monitoring time of the target zone. One of the key approaches to save the sensor battery and prolong the network lifetime is to develop and design data reduction techniques.

In this chapter, we take advantages from all data reduction techniques and propose a hybrid and adaptive data collection mechanism, All-in-One, for energy saving in sensor network applications. The idea behind our mechanism is to make the sensor self-reconfigurable by deciding about the most suitable data reduction technique to be applied according to several parameters, e.g. data redundancy ratio and remaining battery level. Basically, All-in-One works on three phases; the first phase is called on-period and aims to reduce the amount of data transmitted from each sensor either by applying aggregation, compression or prediction techniques. The second phase is called in-period and allows to adapt the sensor data transmission according to the variation of the monitored condition; in-period is based on two data reduction techniques: on-off transmission and adapting the sensing frequency. The third phase is called in-node and seeks the data correlation among neighboring nodes based on in-network correlation and data clustering

techniques.

The remainder of the chapter is organized as follows. Section 4.2 gives an overview about the framework of All-in-One mechanism. Sections 4.3, 4.4 and 4.5 detail the three phases applied at sensor node and CH levels. Simulation results are discussed in section 4.6. Finally, the conclusion is highlighted in section 4.7.

## 4.2/ An Overview to All-in-One Mechanism

All-in-One mechanism is applied on sensor nodes and CHs, and allows eliminating the redundancy existing in sensor networks. The proposed mechanism adapts the same periodic data collection model under the cluster-based architecture as proposed in previous chapters. Figure 4.1 shows the main phases of the proposed mechanism along with the process of redundancy elimination proposed at each phase. At the sensor node level, our mechanism searches the redundancy among the data collected by a sensor at each period and round respectively. On one hand, we search the similarity among collected data by each sensor; then by using an on-period decision table based on the variation level and the sensor battery level, we select the most adequate data reduction method. Subsequently, we propose data reduction algorithms based on three concepts: prediction, compression, and aggregation. On the other hand, our mechanism searches the in-period redundancy by each sensor at each round. Therefore, an in-period decision table is introduced to consider the similarity between data in the round along with the sensor battery level to decide about the appropriate elimination method, e.g. Sensing Frequency Adaptation (SFA) or On-Off Transmission (OOT). At the CH level, the in-node redundancy among neighboring nodes is investigated in order to reduce the periodic number of packets sent to the sink. Subsequently, the redundancy elimination process is based on the packet types. First, the compressed packets are grouped into clusters then sent the cluster centroids to the sink. Second, the aggregated packets are propagated using an in-network aggregation technique then sending the unsimilar data to the sink. Third, the predicted and off packets are directly forwarded to the sink without any elimination process.

## 4.3/ On-Period Redundancy Elimination Model

In sensor network, the periodic data collection is a fundamental operation in order to understand the behavior of the monitored environment and increase the reliability of the taken decision. However, this collection model produces a high redundancy level among the data that leads to send useless data to the sink and consumes the available energy in the sensor. In order to overcome these problems, researchers have focused on three main reduction approaches to eliminate in-period redundancy at each sensor: aggregation, compression and prediction. In this section, we introduce an efficient technique for each approach, then we propose a new hybrid model for removing the in-period redundancy.

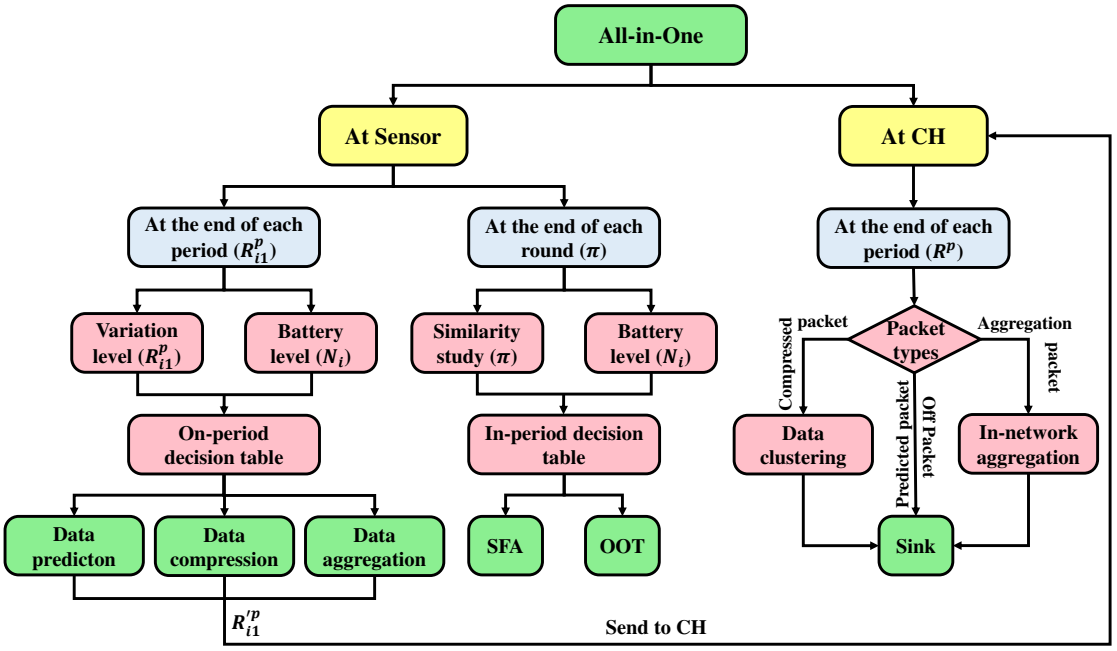



Figure 4.1: Flow diagram of All-in-One mechanism.

### 4.3.1/ Aggregation-Based Reduction Technique

The data aggregation seeks the similarities among the data collected in order to eliminate the existing redundancies and reduce the size of data transmission to the CH. Hence, we first define the $Aggregate$ function that allows each sensor to search the similarities among the readings in $R^p_{i1}$ as follows:

**Definition 4.1** $Aggregate(r^{i1}_j, r^{i1}_k)$. *Assume $r^{i1}_j$ and $r^{i1}_k$ are two readings collected by the same sensor node during a period $p$. Then, $r^{i1}_j$ and $r^{i1}_k$ are considered similar if and only if the difference between them is less than a defined threshold $\epsilon$ as follows:*

$$Aggregate(r^{i1}_j, r^{i1}_k) = |r^{i1}_j - rv_k| \leq \epsilon \tag{4.1}$$

where $\epsilon$ is a user-defined threshold determined according to the application requirements.

Then, in order to maintain the accuracy of the aggregated data, we define the weight, called $wgt$, for each reading as follows:

**Definition 4.2** *The weight of a reading $wgt(r^{i1}_t)$ is defined as the number of similar readings to $r^{i1}_t$ in the same reading set $R^p_{i1}$.*

Based on the $Aggregate$ and weight functions, the sensor node searches the similarity among every pair of readings in $R^p_{i1}$ until no more redundancy exists (Algorithm 4.1). The algorithm takes as input the set of readings collected by a sensor during a period and returns, as output, the aggregated set of readings that will be sent to the CH at the end of the period. For each collected reading, the sensor searches for its similarity (according to the $Aggregate$ function) with all readings in the set; if the two compared readings are similar (according to the similarity threshold) then the weight of the corresponding reading is added by one (lines 2-9). Then, the sensor calculates the weight for each reading and adds it to the aggregated set that will send to the CH (line 10).

**Algorithm 4.1** Data Aggregation Algorithm.

**Require:** A sensor node: $S_{i1}$; A period: $p$; A set of readings: $R^p_{i1}$; similarity threshold: $\epsilon$.
**Ensure:** Aggregated set of readings: $R^{'p}_{i1}$.

1: $R^{'p}_{i1} \leftarrow \emptyset$
2: **for** each reading $r^{i1}_t \in R^p_{i1}$ **do**
3: $wgt(r^{i1}_t) = 1$
4: **for** each reading $r^{i1}_k \in R^p_{i1}$ where $k > t$ **do**
5: **if** $Aggregate(r^{i1}_t, r^{i1}_k) \leq \epsilon$ **then**
6: $wgt(r^{i1}_t) = wgt(r^{i1}_t) + 1$
7: delete $r^{i1}_k$ from $R^p_{i1}$
8: **end if**
9: **end for**
10: $R^{'p}_{i1} \leftarrow R^{'p}_{i1} \cup \{(r^{i1}_t, wgt(r^{i1}_t))\}$
11: **end for**
12: **return** $R^{'p}_{i1}$

### 4.3.2/ Compression-Based Reduction Technique

By definition, the compression is the process of combining redundant readings into a reduced set of records. Indeed, in order to determine the data redundancy, the correlation among the readings should be studied. In this chapter, we focus on the Pearson correlation coefficient (PCC) as one of the metrics that is most used to measure the correlation degree among data sets. Pearson coefficient gives a value between $-1$ and $+1$ where $+1$ (respectively $-1$) indicates a perfect (respectively negative perfect) correlation among the datasets. Mathematically, the Pearson correlation coefficient between two data sets $R^p_{i1}$ and $R^p_{j1}$ is given by to the following equation:

$$Pearson(R^p_{i1}, R^p_{j1}) = \frac{\mathcal{T} \sum_{t=1}^{\mathcal{T}} r^{i1}_k r^{j1}_k - \sum_{t=1}^{\mathcal{T}} r^{i1}_i \sum_{t=1}^{\mathcal{T}} r^{j1}_j}{\sqrt{\mathcal{T} \sum_{t=1}^{\mathcal{T}} r^{i1\,2}_i - (\sum_{t=1}^{\mathcal{T}} r^{i1}_i)^2} \sqrt{\mathcal{T} \sum_{t=1}^{\mathcal{T}} r^{j1\,2}_j - (\sum_{t=1}^{\mathcal{T}} r^{j1}_j)^2}} \tag{4.2}$$

where $r^{i1}_t \in R^p_{i1}$, $r^{i1}_t \in R^p_{j1}$ and $\mathcal{T}$ is the number of readings in $R^p_{i1}$ or $R^p_{j1}$.

Therefore, $R^p_{i1}$ and $R^p_{j1}$ are considered to be highly correlated (e.g. redundant) if and only if:

$$|Pearson(R^p_{i1}, R^p_{j1})| > \rho \tag{4.3}$$

where $\rho \in [0, 1]$ is the Pearson's threshold.

Algorithm 4.2 shows the compression technique applied over the data collected by each sensor node during a period, based on the Pearson coefficient metric. First, all the readings are assumed to be correlated and $R^p_{i1}$ is assigned to a temporary set of reading subsets, e.g. $T$ (line 2). Then, the correlation among the readings is calculated by dividing them into two equal subsets using the function *Partition* (line 4). Thus, if the

correlation exceeds the Pearson threshold then the readings are considered redundant and, consequently, the average of the readings is computed (e.g. $\overline{r_t^{i1}}$) and added with its weight (e.g. $wgt(\overline{r_t^{i1}})$) to the final reading set that will sent to the CH (lines 5-10). Otherwise, e.g. the absolute value of the correlation coefficient does not exceed the Pearson threshold, the readings are considered unsimilar and we repeat the process over each subset until all readings within each subset become redundant. Therefore, at the end of each period, each sensor will send the compressed set of readings $R_{i1}^{'p}$ to the CH.

---

**Algorithm 4.2** Data Compression Algorithm.

---

**Require:** A sensor node: $S_{i1}$; A period: $p$; A set of readings: $R_{i1}^{p}$; Pearson threshold: $\rho$.
**Ensure:** Compressed set of readings: $R_{i1}^{'p}$.

1: $R_{i1}^{'p} \leftarrow \emptyset$
2: $T \leftarrow R_{i1}^{p}$
3: **for** each set $R_{k1}^{p} \in T$ **do**
4: $(R_{k1_l}^{p}, R_{k1_r}^{p}) = Partition(R_{k1}^{p})$
5: **if** $Pearson\,(R_{k1_l}^{p}, R_{k1_r}^{p}) \leq \rho$ **then**
6: $\overline{r_t^{i1}} = Mean(R_{k1}^{p})$
7: $wgt(\overline{r_t^{i1}}) = |R_{k1}^{p}|$
8: // $|R_{k1}^{p}|$ is the total number of elements in $R_{k1}^{p}$
9: $R_{i1}^{'p} \leftarrow R_{i1}^{'p} \cup \{(\overline{r_t^{i1}},\ wgt(\overline{r_t^{i1}}))\}$
10: remove $R_{k1}^{p}$ from $T$
11: **else**
12: $T \leftarrow T \cup \{R_{k1_l}^{p}, R_{k1_r}^{p}\}$
13: remove $R_{k1}^{p}$ from $T$
14: **end if**
15: **end for**
16: **return** $R_{i1}^{'p}$

---

### 4.3.3/ PREDICTION-BASED REDUCTION TECHNIQUE

In sensing-based applications, the data prediction allows each sensor to build, based on the collected data, a predictive model in order to send to the sink which, in its turn, regenerates the raw data. In this work, we used the same prediction model based on the Newton Forward Differences (NFD) method proposed in section 2.2.1 in chapter 2. Such model takes the periodic data collected by a sensor, e.g. $R_{i1}^{p}$, and finds the polynomial coefficient set, e.g. $R_{i1}^{'p}$, based on the forward difference table (FDT) and the Newton's formula. Finally, in order to allow the sink to regenerate the readings in $R_{i1}^{p}$, the sensor must send the set of variables, e.g. $R_{i1}^{'p}$, needed in the NFD formula to calculate the $r_t^{i1}$ values of all readings.

### 4.3.4/ Performance Discussion of On-Period Techniques

This section gives further considerations of the three introduced on-period techniques by studying the thresholds' selection, the accuracy, the complexity, and the energy consumption.

#### 4.3.4.1/ Selection of Thresholds' Values

Obviously, the efficiency of the aggregation, compression, and prediction techniques are highly related to the selection of the thresholds $\epsilon$, $\rho$, and $d$ respectively. Subsequently, increasing or decreasing the threshold values may change the performance of several metrics in sensor networks, such as: the accuracy, the data latency, the data transmission ratio, and the energy consumption. Hence, selecting the appropriate values of thresholds are critical in the first stage of our mechanism. Therefore, we consider that the thresholds' values should be determined by the decision makers or experts depending on the application requirements. For instance, in health monitoring applications, the thresholds should optimize the accuracy of the collected data more than other metrics, while, in the environmental applications, the energy conservation gets the highest priority compared to other metrics. Thus, these parameters are based on the application criticality and the studied phenomenon.

After selecting their values, the decision makers assign the thresholds accordingly into all sensor nodes prior to deployment or they can adjust it online in function of the application requirement.

#### 4.3.4.2/ Accuracy Study

In compression-based and prediction-based techniques, the increase of the values of thresholds (e.g. $d$ and $\rho$) will proportionally increase the amount of data sent, thus the accuracy of the information sent, and vice versa. While, in the aggregation-based technique, the accuracy of the sent data will increase with the decrease of similarity threshold, e.g. $\epsilon$. However, in our mechanism, the $wgt$ function defined in the aggregation and compression algorithms (e.g. Algorithms 4.1 and 4.2) will maintain the full accuracy of the sent data.

#### 4.3.4.3/ Complexity Study

The complexity is an important metric in sensor networks due to the limited sensor resources, especially processing and storage. From one hand, the processing complexity of any proposed technique may affect the system latency which is crucial in many sensor applications, especially in healthcare or military. On the other hand, sensors are characterized by relatively small memory size; hence, any technique should satisfy the memory constraint. The complexity of the three proposed techniques can be studied as follows:

- The aggregation technique: each sensor node $S_{i1}$ forms a set $R_{i1}^{p}$ of $\mathcal{T}$ readings in each period. Due to the $Aggregate$ function, the size of this set can be reduced from $\mathcal{T}$ to $|R_{i1}^{'p}|$. Therefore, this technique has at most $O(|R_{i1}^{'p}|^2)$ as a computation

complexity at the sensor and saves at most $2 \times |R_{i1}^{'p}|$ values, e.g. readings with their weights, at each period in its memory.

- The compression technique: the sensor $S_{i1}$ recursively divides the reading set collected in every period into two equal partitions before calculating their correlation according to Pearson coefficient. Hence, the computation complexity of the compression algorithm will be of $O(|R_{i1}^{p}| \times log(|R_{i1}^{p}|))$ while the memory storage will be equal, at most, to $2 \times |R_{i1}^{'p}|$ (e.g. mean values with their weights), similarly to that of the aggregation technique.

- The prediction technique: according to the NFD method, the sensor sends the set of coefficients, e.g. $R_{i1}^{'p}$, calculated at each period to the CH. Thus, the computation complexity of the prediction algorithm should be of $O(d \times log(|R_{i1}^{p}|))$ while the memory storage is limited to the length of the NFD coefficient set, e.g. $|R_{i1}^{'p}|$.

Based on the above study, we clearly show that the complexities of all techniques are suitable for the case of sensor nodes.

#### 4.3.4.4/ Energy Consumption Study

In sensor networks, the data transmission operation consumes most of the sensor energy compared to other operations, e.g. sensing and processing [149]. Thus, minimizing the periodic data transmitted from sensors and CHs is mandatory to save the energies. Hence, the three proposed on-period techniques can be considered as good solutions for conserving the node energies and extending their lifetime. This is due to the redundancy elimination process introduced in each one of them that allows to reduce the amount of transmitted data and only send the useful information towards the sink. Furthermore, as mentioned before, the elimination process ensures the accuracy of the sent data and limits the effect on the decision made by the end user.

### 4.3.5/ Hybrid-Based On-Period Reduction Technique

Indeed, the selection among the data reduction approaches (aggregation, compression or prediction) is a crucial decision for the sensor since it affects several performance metrics. For instance, the data prediction technique can highly save the sensor's energy because it reduces the data transmission more than aggregation and compression techniques. However, the prediction technique can negatively affect the accuracy of the transmitted data. Hence, we propose a hybrid-based on-period reduction model that takes advantages from several reduction techniques while optimizing several performance metrics. The proposed model is based on two main parameters, e.g. the condition variation and the remaining sensor battery, in order to decide the reduction technique that should be used in each period. Subsequently, the condition variation is calculated according to the ANOVA and a statistical test, e.g. Bartlett test.

#### 4.3.5.1/ ANOVA Model and Bartlett Test

ANOVA is a well-known statistical method that is used to test the variance among a group of data sets if it is significant or not. First, ANOVA computes a $T$-statistic value, according to a statistical test, then the data sets are considered redundant (or have low variance) if the calculated $T$ is less than a critical value $T_\alpha$ for some false-rejection probability $\alpha$; more the value of $T_\alpha$ is decreased, more the redundancy among the data sets is.

On the other hand, Bartlett test [150] checks if a group of data sets have an equal variance. Thus, Bartlett test verifies the null hypothesis that variances are equal across data sets comparing to the alternative hypothesis that the variances are significant. In our case, the objective is to calculate the variance among readings collected by a sensor during a period (e.g. $R_{i1}^p$). Hence, we first divide $R_{i1}^p$ into $\mathcal{V}$ equal divisions (or subsets) where each division $\mathcal{V}_j$, $j \in [1, \mathcal{V}]$, contains $\mathcal{T}/\mathcal{V}$ readings. Then, the Bartlett test can be applied over $R_{i1}^p$ as follows:

$$T = \frac{(\mathcal{T} - \mathcal{V}) \ln(\sigma_p^2) - \left(\frac{\mathcal{T}}{\mathcal{V}} - 1\right) \sum_{j=1}^{\mathcal{V}} \ln(\sigma_j^2)}{\lambda} \tag{4.4}$$

where :

$$\lambda = \frac{3 \times \mathcal{T} - 2 \times \mathcal{V} + 1}{3 \times (\mathcal{T} - \mathcal{V})}$$

and $\sigma_p^2$ is the pooled variance that is defined as:

$$\sigma_p^2 = \frac{1}{\mathcal{T} - \mathcal{V}} \sum_{j=1}^{\mathcal{V}} \sigma_j^2$$

Therefore, in order to test the variance $T$ among the readings in $R_{i1}^p$, we select two critical values for $T_\alpha$, e.g. $T_{\alpha_0}$ and $T_{\alpha_1}$ where $\alpha_0 < \alpha_1$. Then, the condition variation is based on:

- $T \leq T_{\mathcal{V}-1,\alpha_0}$ or *low variation*: the variance among the divisions is not significant and the readings in $R_{i1}^p$ are considered similar.
- $T_{\mathcal{V}-1,\alpha_0} < T \leq T_{\mathcal{V}-1,\alpha_1}$ or *medium variation*: the variance among the divisions is a bit significant and the readings in $R_{i1}^p$ are considered redundant.
- $T > T_{\mathcal{V}-1,\alpha_1}$ or *high variation*: the variance among the divisions is significant.

#### 4.3.5.2/ Sensor Battery Level

The lifetime of the sensor networks is heavily related to the sensor battery level which, in its turn, can be quickly consumed when the amount of data transmission increases. Hence, in addition to condition variation level, we propose to take into account the remaining energy of the sensor in order to adapt the periodic data transmission to the CH. The idea is that when the sensor battery level becomes crucial, e.g. less than a defined threshold, its data transmission must be more and more reduced but without highly affecting the data integrity.

Let assume that the initial energy of the sensor node is $E_i$ and the remaining one during the current period $p$ is $E_r$. Then, we define a critical threshold $E_c$ where the sensor energy becomes crucial if it reaches this threshold. Therefore, the decision about the sensor battery level during a period $p$ can be made as follows:

- if $E_i \geq E_c$ then *high battery level*.
- otherwise, *low battery level*.

#### 4.3.5.3/ On-Period Data Decision

At the end of each period, the hybrid-based reduction technique allows each sensor to decide about the reduction approaches (aggregation, compression and prediction) that should be applied over the collected data. Table 4.1 shows the decision made by the sensor based on the calculated variation and battery levels. Subsequently, the selection of the reduction approaches inside the on-period data decision table is motivated by the following reasons:

- if the variation and battery levels are low then the data prediction must be used. This will reduce the data transmission to the minimum (thus save the sensor energy) but without losing the information collected by the sensor.
- if the variation is high then the data aggregation is preferably to be used. This is because the aggregation will decrease the similarity between the transmitted data without ensuring a high level of data accuracy.
- otherwise, the data compression constitutes an ideal technique that compromises between data reduction and data accuracy.

| Variation level / Battery level | *Low* | *High* |
|---|---|---|
| *Low* | Data prediction | Data compression |
| *Medium* | Data compression | Data compression |
| *High* | Data aggregation | Data aggregation |

Table 4.1: On-period data decision table.

## 4.4/ In-Period Redundancy Elimination Model

Mostly, the data collected by each sensor during successive periods are highly correlated depending on the variation of the monitored condition. Particularly, the slowdown of the environment leads to increase the redundancy among the sensed data which results in sending useless data to the sink and consuming the sensor energy. Hence, eliminating the in-period data redundancy becomes an essential technique to achieve fair data reduction rates and conserve the limited energy resources of sensor networks. In the next section, we introduce two mechanisms in order to search, then eliminate, the redundancy existing among periods: on-off transmission and sensing frequency adaptation.

### 4.4.1/ Sensing Frequency Adaptation (SFA) Mechanism

In the periodic collection model, the selection of the appropriate sensing frequency of each sensor is a very important decision before deploying the network. Consequently, a high sensing frequency can lead to increase the redundancy among the collected data and consume the sensor energy while the decreasing of the sensing frequency can affect the accuracy of the transmitted data. Hence, adapting the sensing frequency to the environment variation is thereby resulting in data reduction and saving sensor energy.

Mathematically, let assume a round $\pi$ consisting of $P$ period in which a sensor node $N_i$ will collect a set of readings sets as follows: $R_i = \{R_{i1}^1, R_{i1}^2, \ldots, R_{i1}^P\}$. Therefore, in order to study the condition variation, ANOVA and Bartlett test are applied again over the data sets in $R_i$. Thus, the condition is "slow down" if the calculated variation $T$ is less than a certain threshold $T_{P-1,\beta}$ for some false rejection probability (risk $\beta$). Consequently, the sensor must adapt its sensing frequency according to the $Adapting$ function based on the Bezier curve [151]:

$$Adapting(T, T_{P-1,\beta}, C_r, \mathcal{T}) = \begin{cases} \frac{(\mathcal{T}-2b_y)}{4b_x^2}T^2 + \frac{b_y}{b_x}T & if\ (T_{P-1,\beta} - 2b_x = 0) \\ (\mathcal{T} - 2b_y)(\propto (T))^2 + 2b_y \propto (T), & if\ (T_{P-1,\beta} - 2b_x \neq 0) \end{cases}$$

where

$$\propto (T) = \frac{-b_x + \sqrt{{b_x}^2 - 2b_x \times T + T_{P-1,\beta} \times T}}{T_{P-1,\beta} - 2b_x} \ \wedge \begin{cases} 0 \leq b_x \leq T_{P-1,\beta} \\ 0 \leq T \leq T_{P-1,\beta} \\ T_{P-1,\beta} > 0 \end{cases}$$

and $b_x = -T_{P-1,\beta} \times C_r + T_{P-1,\beta}$ while $b_y = \mathcal{T} \times C_r$.

Subsequently, the $Adapting$ function takes four variables as input: the variance between readings in a round ($T$), the variance threshold ($T_{P-1,\beta}$), the criticality of the monitored application ($C_r$) and the original period size ($\mathcal{T}$). Indeed, the application criticality ($C_r$) is a value between $0$ and $1$ that is assigned by the expert depending on the monitored application and that should be taken into account when adapting the sensor frequency. For instance, $C_r$ must take a value near to $1$ in high critical applications (i.e. healthcare and military) and near to $0$ in low critical applications (i.e. weather and environment monitoring). Therefore, the $Adapting$ function calculates the new sensing frequency of the sensor in the next round.

### 4.4.2/ On-Off Transmission (OOT) Mechanism

The objective of this technique is to avoid sending similar data in successive periods from each sensor to the CH. Thus, the sensor will update the CH about the condition variation only if a noticed difference is detected compared to the last sent data. This will decrease the number of packets sent from each sensor, save its energy and reduce the congestion in the network. Indeed, one can find several functions that allows to search the similarity among data sets such as Jaccard, Dice, Cosine, etc. In this chapter, we focus on the Jaccard similarity as one of most used and well adapted functions to several domains. For the sake of simplicity, let assume a round consisting of two periods, e.g. $R_i = \{R_{i1}^1, R_{i1}^2\}$, thus reading sets in $R_i$ are considered similar according to the Jaccard function if:

$$Jaccard(R_{i1}^1, R_{i1}^2) = \frac{|R_{i1}^1 \cap R_{i1}^2|}{|R_{i1}^1 \cup R_{i1}^2|} \geq t_J \tag{4.5}$$

where $t_J$ is the Jaccard threshold in $[0, 1]$ where $0$ indicates that the readings are totally different and $1$ that are totally equal.

Algorithm 4.3 shows the on-off transmission mechanism applied at each sensor during a round. Indeed, we define two types of packets that will send by the sensor: $On_Packet$ which contains the identification (id) of the sensor with its readings collected during the current period; $Off_Packet$ which only contains the id of the sensor informing the CH that the current collected readings are removed due to the similarity with the previous ones. Thus, the sensor sends the reading set collected during the first period to the CH in a $On_Packet$ while saving it in its memory at the same time (lines 1-2). Then, for every new reading set collected in the next period, the sensor searches its similarity with the set saved in the memory based on the Jaccard function; if the new set is similar to the saved one, then the sensor removes the new one, while sending a $Off_Packet$ to the CH (lines 4-6). Otherwise, e.g. the new one is not similar to the saved one, the sensor sends the new reading set to the CH while replacing the saved set by the new reading set (lines 7-10).

**Algorithm 4.3** On-Off Transmission Algorithm.

**Require:** A sensor node: $N_i$, a round: $\pi$, set of reading sets: $R_i = \{R_{i1}^1, R_{i1}^2, \ldots, R_{i1}^P\}$, Jaccard similarity threshold: $t_J$.
**Ensure:** Saved reading set: $R_{i1}^j$.

1: $R_{i1}^j \leftarrow R_{i1}^1$
2: $On_Packet(i, R_{i1}^j)$
3: **for** each set $R_{i1}^k \in R_i$ where $k \geq 2$ **do**
4: **if** $Jaccard\ (R_{i1}^k, R_{i1}^j) \geq t_J$ **then**
5: ignore $R_{i1}^k$
6: $Off_Packet(i)$
7: **else**
8: $R_{i1}^j \leftarrow R_{i1}^k$
9: $On_Packet(i, R_{i1}^j)$
10: **end if**
11: **end for**
12: **return** $R_{i1}^j$

### 4.4.3/ Hybrid-Based In-Period Reduction Technique

Obviously, SFA and OOT can both minimize the in-period data redundancy and save the sensor energy. However, SFA can reduce the data transmission to the CH more than OOT because it minimizes its data collection even all readings collected in successive periods are similar. Otherwise, OOT can ensure more data accuracy than SFA because just very similar data collected are not sent to the CH. Hence, in order to make a trade-off between energy saving and data accuracy, we propose a hybrid-based in-period model

that allows each sensor to select between SFA and OOT at the end of each round. The proposed model takes into account the in-period similarity among the collected data and the remaining sensor battery then it decides about the suitable technique to apply at the end of each round. Subsequently, the sensor battery level usage is similar to the situation proposed in subsection 4.3.5.2 while the in-period similarity study is described on the next section.

#### 4.4.3.1/ IN-PERIOD SIMILARITY STUDY

Indeed, similarity functions are one of the most accurate approaches to search the redundancy among the data compared to other approaches, particularly ANOVA and distance functions. Therefore, we propose to use the Jaccard similarity function in order to determine the similarity level among data collected in successive periods. Once the data similarity level is calculated, the sensor decides about the in-period technique that must be used according to the in-period decision table (see next section). Given a round $\pi$ consisting of two periods, e.g. $R^1_{i1}$ and $R^2_{i1}$, the Jaccard similarity between both periods can be calculated according to the equation 4.5. Then, in our model, we distinguish between three levels of similarities among data collected in $\pi$:

- $0 \leq Jaccard(R^1_{i1}, R^2_{i1}) \leq 0.5$ or *low similarity*: this indicates that the monitored condition is rapidly changing over the periods.
- $0.5 < Jaccard(R^1_{i1}, R^2_{i1}) \leq 0.75$ or *medium similarity*: this indicates that the monitored condition is slowly changing over the time which leads to a certain level of redundancy among the collected data.
- $0.75 < Jaccard(R^1_{i1}, R^2_{i1}) \leq 1$ or *high similarity*: in which the monitored condition is not significantly changing which results in a high similarity among the collected data.

#### 4.4.3.2/ IN-PERIOD DECISION TABLE

The in-Period Decision table shows the decision made by the sensor at the end of each round based on the data similarity and the battery levels (Table 4.2). Subsequently, the sensor selects the in-period reduction technique according to the following criteria:

- The sensor must decrease its sensing frequency when the similarity level increases, either with low or high battery level. This will reduce the redundancy among the collected data.
- By fixing to the similarity level to low, medium or high, the sensor must decrease it sensing frequency with the decreasing level of its battery. This will save the sensor energy and avoid a rapid depletion of its battery.
- If a high data similarity level is detected, the sensor will not send the current collected data to the CH (e.g. apply OOT) and will adapt its sensing frequency to the minimum.

| Similarity level / Battery level | *Low* | *High* |
|---|---|---|
| *Low* | $\mathcal{T}^{'} = 40\%$ of $\mathcal{T}$ | $\mathcal{T}^{'} = \mathcal{T}$ |
| *Medium* | $\mathcal{T}^{'} = 30\%$ of $\mathcal{T}$ | $\mathcal{T}^{'} = 60\%$ of $\mathcal{T}$ |
| *High* | OOT + $\mathcal{T}^{'} = 20\%$ of $\mathcal{T}$ | OOT + $\mathcal{T}^{'} = 40\%$ of $\mathcal{T}$ |

Table 4.2: In-period data decision table.

## 4.5/ In-Node Redundancy Elimination Model

At the end of each period, the CH receives all data sets coming from its sensors. Indeed, such data are mostly redundant due to the spatial and temporal correlation among the sensors. Therefore, the CH can remove this redundancy in order to reduce the number of packets sent to the sink (thus saves its own energy) and provide only a useful information to the end user. In this section, we introduce two approaches to eliminate in-node (e.g. between nodes) redundancy at the CH: in-network aggregation and data clustering. Subsequently, in order to apply each of the proposed approaches, the CH must recalculate the raw data, e.g. $R^p_{i1}$, of each received data set, e.g. $R^{'p}_{i1}$, according to the applied in-period approaches.

### 4.5.1/ In-Network Aggregation Approach

This approach aims to eliminate redundant data sets generated by pairs of neighboring sensors before sending to the sink. Pairs of redundant sets are determined by using distance functions that compute the dissimilarities between two data sets. Thus, two data sets are considering duplicate if the distance between them is less than a predefined threshold. Once all duplicated pairs are found, the CH selects a subset of data to send to the sink while eliminating the other ones. Therefore, the in-network aggregation approach is divided into two steps:

- Pairs generation: In this step, the CH searches all pairs of redundant data sets based on the distance functions. In this chapter, we use the Euclidean distance as one of the most distance functions used in the literature. Given two sets of data $R^p_{i1}$ and $R^p_{j1}$ collected by two sensors at the same period $p$, then the Euclidean distance $E_d$ between both sets is:

$$E_d(R^p_{i1}, R^p_{j1}) = \sqrt{\sum_{t=1}^{\mathcal{T}} (r^{i1}_t - r^{j1}_t)^2} \tag{4.6}$$

  where $r^{i1}_t \in R^p_{i1}$ and $r^{i1}_t \in R^p_{j1}$. Then, $R^p_{i1}$ and $R^p_{j1}$ are considered redundant if the Euclidean distance between them is less than a threshold, $t_E$:

$$E_d(R^p_{i1}, R^p_{j1}) \leq t_E \tag{4.7}$$

- Pairs selection: After determining all redundant pairs, the CH tries to reduce the number of data sets to the sink by selecting a subset among them instead of sending the whole data sets (Algorithm 4.4). For each generated pair, the CH selects

the received set having the highest number of elements, e.g. $|R^{'p}_{j1}|$, then it adds it to the final list of data sets that will send to the sink (line $2-4$). Simultaneously, the CH removes all pairs that contain $R^{'p}_{i1}$ or $R^{'p}_{i1}$ from the set of generated pairs (line $5$).

**Algorithm 4.4** In-Network Aggregation Algorithm.

**Require:** List of generated pairs: $A = \{(R^p_{i1}, R^p_{j1})$ such that $E_d(R^p_{i1}, R^p_{j1}) \leq t_E$ and $i \neq j\}$.
**Ensure:** List of sent data sets: $L$.
1: $L \leftarrow \emptyset$
2: **for** each pair $(R^p_{i1}, R^p_{j1}) \in A$ **do**
3: Consider $|R^{'p}_{i1}| \geq |R^{'p}_{j1}|$ // $|.|$ indicates the length of the reading set
4: $L \leftarrow L \cup \{R^{'p}_{i1}\}$
5: Remove all pairs containing $R^p_{i1}$ or $R^p_{j1}$
6: **end for**
7: **return** $L$

### 4.5.2/ Data Clustering Approach

Generally, clustering is a data exploratory task that aims to group data into a set of $K$ clusters in a way that the similarity among data in the same cluster is high and that among clusters is low. Thus, data clustering can be an efficient solution to reduce the data transmission from the CH by sending only one information, e.g. the centroids of the clusters, from each cluster to the sink. Researchers have proposed a lot of clustering techniques for various types of data. One of the most popular algorithms in data clustering is K-means [152]; it is flexible, simple, already adapted to huge number of applications and used with various kinds of data [153, 154, 155].

Typically, the K-means is an iterative algorithm in which the process starts by randomly selecting an initial centroid for each cluster. Then, each data set is assigned to the nearest centroid, according to the Euclidean distance (see equation 4.6), and the first round of cluster formation is performed. After that, the cluster centroids are updated and the process is repeated until the convergence of the criterion function (Algorithm $4.5$).

**Algorithm 4.5** K-means Algorithm.

**Require:** Set of reading sets: $R^p = \{R^1_{i1}, R^2_{i1}, \ldots, R^n_{i1}\}$, cluster number: $K$.
**Ensure:** Set of clusters $C = \{C_1, C_2, \ldots, C_K\}$.
1: **for** $j \leftarrow 1$ to $K$ **do**
2: randomly choose centroid $c_j$ among $R^p$ belongs to $C_j$
3: **end for**
4: **repeat**
5: **for** each data set $R^p_{i1} \in R^p$ **do**
6: Assign $R^p_{i1}$ to the cluster $C_j$ with nearest $c_j$
(i.e., $E_d(R^p_{i1}, R^p_{j1*}) \leq E_d(R^p_{i1}, R^p_{j1})$; $j \in \{1, \ldots K\}$)
7: **end for**
8: **for** each cluster $C_j, where\ j \in \{1, \ldots K\}$ **do**
9: Update the centroid $c_i$ to be the centroid of all data readings currently in $C_j$

10: **end for**
11: **until** no change in the cluster memberships
12: return $C$

### 4.5.3/ HYBRID-BASED IN-NODE REDUCTION TECHNIQUE

Obviously, in-network aggregation and data clustering approaches are quite different from the redundancy elimination point of view. Thus, they have different impacts regarding various performance metrics, especially number of periodic packets sent and data accuracy. Since the first approach searches the redundant data sets in pairs instead of groups in the second one, it saves the data integrity more than the other one. However, the data clustering saves the sensor energy more than the in-network aggregation because it limits the number of transmitted packets to the cluster centroids. Thus, in order to ensure a trade-off between both metrics, we propose a hybrid in-node reduction approach to apply over the data sets received by the CH at each period.

Let first recall the four types of packets received by a CH during a period:

- $Off_Packet$ indicating that the data set collected at the current period is similar to that sent in the previous one.
- $Aggregate_Packet$ containing the data aggregated according to the Algorithm 4.1.
- $Compressed_Packet$ containing the data compressed according to the Algorithm 4.2.
- $Predicted_Packet$ containing the coefficient set calculated based on the Newton forward formula.

Therefore, the forwarded packets from the CH to the sink can be shown according to the in-node decision algorithm (Algorithm 4.6). First, all packets of types $Off_Packet$ and $Predicted_Packet$ will be added to the final list of sets sent to the sink, e.g. $I$ (lines 4-7). Indeed, such types of packets do not consume the energy of CH because they contain no data (e.g. $Off_Packet$) or a few data values (coefficient set in $Predicted_Packet$). Then, for the sensors sending aggregated packets, the CH applies the in-network aggregation approach in order to remove the redundancy among them and reduce the number of packets sent to the sink. Finally, the CH applies the K-means algorithm to the data sets compressed by the sensors (lines 10-12 and 16).

**Algorithm 4.6** In-Node Reduction Algorithm.

**Require:** Set of reading sets: $R^p = \{R^1_{i1}, R^2_{i1}, \ldots, R^n_{i1}\}$, cluster number: $K$, Euclidean distance threshold: $t_E$.
**Ensure:** Final list of sent packets: $I$.
1: $I \leftarrow \emptyset$
2: $A \leftarrow \emptyset$
3: $C \leftarrow \emptyset$
4: **for** each $R^p_{i1} \in R^p$ **do**
5: **if** $R^p_{i1}$ is of type $Off_Packet$ or $Predicted_Packet$ **then**
6: $I \leftarrow I \cup \{R^p_{i1}\}$

```
    else
      if $R_{i1}^{p}$ is of type $Aggregate\_Packet$ then
        $A \leftarrow A \cup \{R_{i1}^{p}\}$
      else
        $C \leftarrow C \cup \{R_{i1}^{p}\}$
      end if
    end if
  end for
  $I \leftarrow I \cup$ In-Network_Aggregation$(A, t_E)$
  $I \leftarrow I \cup$ Data_Clustering$(C, K)$
  return $I$
```

## 4.6/ Simulation Results

We evaluated the performance of our mechanism using the same real sensor data collected from Intel Berkeley Research Lab [145] with the same scenario adapted in the second chapter. We implemented the algorithms used in our mechanism based on Java simulator and we compared the obtained results to those obtained in the PFF [61] and S-LEC [70].

Table 4.3 summarizes the parameters used in our simulation with their tested values.

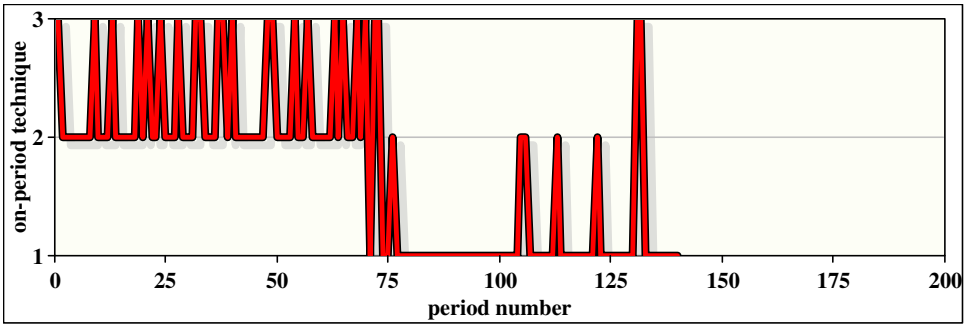
| Parameter | Symbol | Values |
|---|---|---|
| *Aggregate* threshold | $\epsilon$ | 0.05, 0.1, 0.2 |
| Pearson threshold | $\rho$ | 0.4, 0.5, 0.6, 0.7 |
| Prediction threshold | $d$ | 4, 5, 6 |
| Period size | $\mathcal{T}$ | 50, 100, 250 |
| ANOVA thresholds | $\alpha_0$, $\alpha_1$ | 0.01, 0.05 |
| Initial sensor energy | $E_i$ | 5 mJ |
| Critical energy threshold | $E_c$ | $\frac{E_i}{2}$ |
| Round size | $\pi$ | 2 periods |
| Jaccard threshold | $t_J$ | 0.7 |
| Eulidean distance threshold | $t_E$ | 0.4 |
| Clusters number | $K$ | 4, 6, 8 |

Table 4.3: Simulation environment.

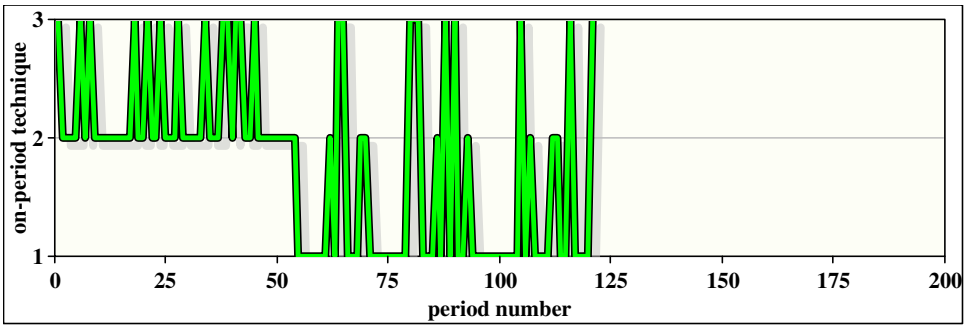
### 4.6.1/ On-Period Decision Study

Figure 4.2 shows which on-period technique has been selected by a sensor at the end of each period based on the on-period decision table. In each subfigure (4.2(a), 4.2(b) and 4.2(c)) represents prediction, compression and aggregation techniques respectively. The obtained results confirm the behavior of our proposed technique as follows: 1) when its remaining energy is high, the sensor selects between compression and aggregation in order to ensure a high data accuracy along with the reduced amount of data transmission; 2) when its remaining energy becomes low, the sensor applies the prediction

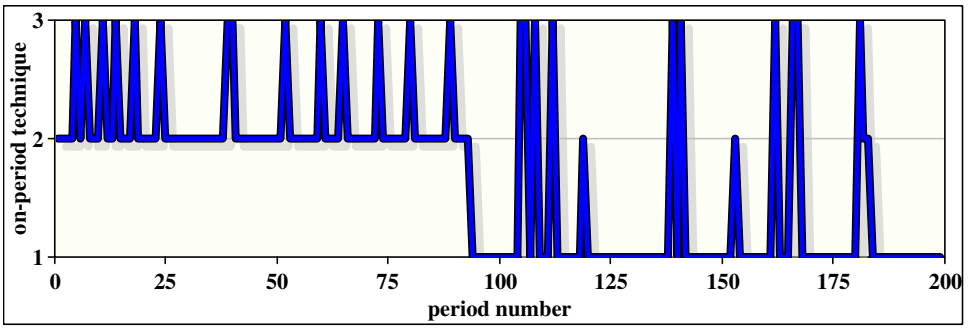
technique, except if the data redundancy is low, in order to reduce to the minimum its data transmission while saving the information integrity. We can also observe that the lifetime of the sensor is more extended with the light physical parameter compared to temperature and humidity; this indicates that the light readings are highly redundant compared to other ones thus the sensor can more reduce its data transmission by applying either compression or prediction techniques.




(a) temperature




(b) humidity




(c) light

Figure 4.2: Variation of the on-period technique selected by the sensor at each period, $\mathcal{T} = 50$, $\epsilon = 0.1$, $\rho = 0.5$, $d = 5$.

### 4.6.2/ IN-PERIOD DECISION STUDY

Figure 4.3 shows the decision made by the sensor at the end of each round according to the in-period decision table. Subsequently, the numbers in the y-axes are describing as follows: 1, 3 and 5 indicate a low battery level with low, medium and high data similarity respectively; 2, 4, and 6 indicate a high battery level with low, medium and high data similarity respectively. The obtained results reveal several observations: 1) the sensing frequency of the sensor is dynamically adapted after each round in each of the three conditions (temperature, humidity and light). 2) By analyzing the new sensing frequencies of the sensors, we observe that the light condition reduces its data collection more than the other conditions because the light readings are more similar compared to other ones. Hence, we observe that the light sensor, mostly, selects between the fifth and sixth in-period techniques depending on its battery level, e.g. low or high. Otherwise, the data similarity level of temperature readings almost varies between low and medium, thus its sensing frequency varies between 1 and 4, while the humidity readings are more redundant than temperature and it varies between 1 and 5.

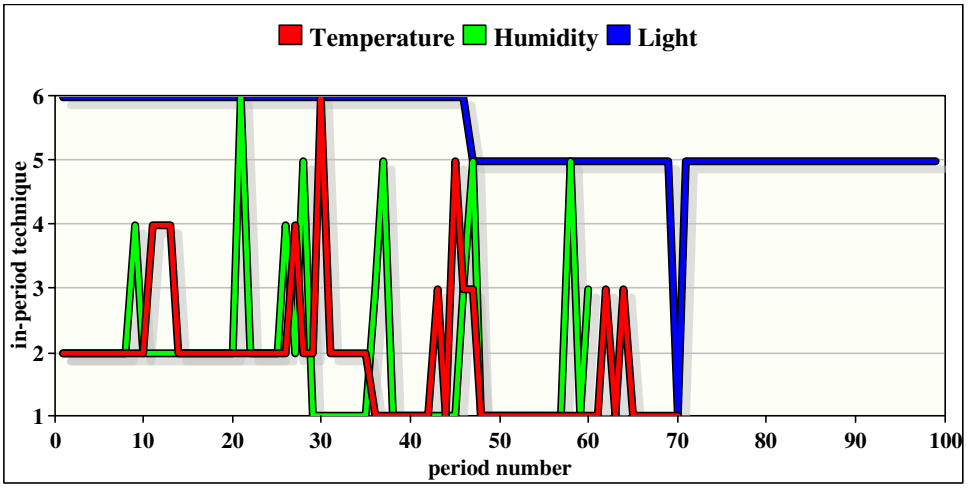



Figure 4.3: Variation of the in-period decision made by the sensor at each round, $\mathcal{T} = 50$, $\epsilon = 0.1$, $\rho = 0.5$, $d = 5$.

Based on the selected in-period technique, Figure 4.4 shows the new sensing frequencies of a sensor after adapting its sampling rate after each round. Because the light readings are very similar, the light sensor adapts its sensing frequencies to the minimum in order to avoid collecting redundant data, e.g. 40% when its battery level is low and 20% when its battery level is high. On the other hand, the temperature and humidity readings are less similar than those of light, thus they adapt their sensing frequencies less than the light sensor, e.g. mostly between 20% and 50% for the temperature and between 10% and 50% for the humidity.

### 4.6.3/ DATA TRANSMISSION RATIO AT SENSOR

Figure 4.5 shows the number of readings sent from each sensor to the CH after applying both on-period and in-period techniques, for 15 periods of simulations. The results are dependent on the period size (Fig. 4.5(a)), the aggregate threshold (Fig. 4.5(b)), the compression threshold (Fig. 4.5(c)) and the prediction polynomial degree (Fig. 4.5(d)).

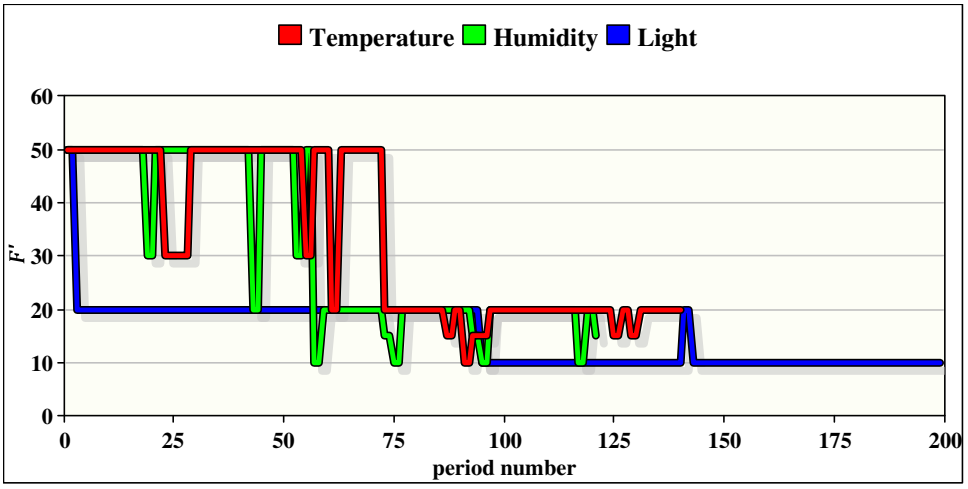



Figure 4.4: Variation of the sensing frequency of a sensor during periods, $\mathcal{T} = 50$, $\epsilon = 0.1$, $\rho = 0.5$, $d = 5$.

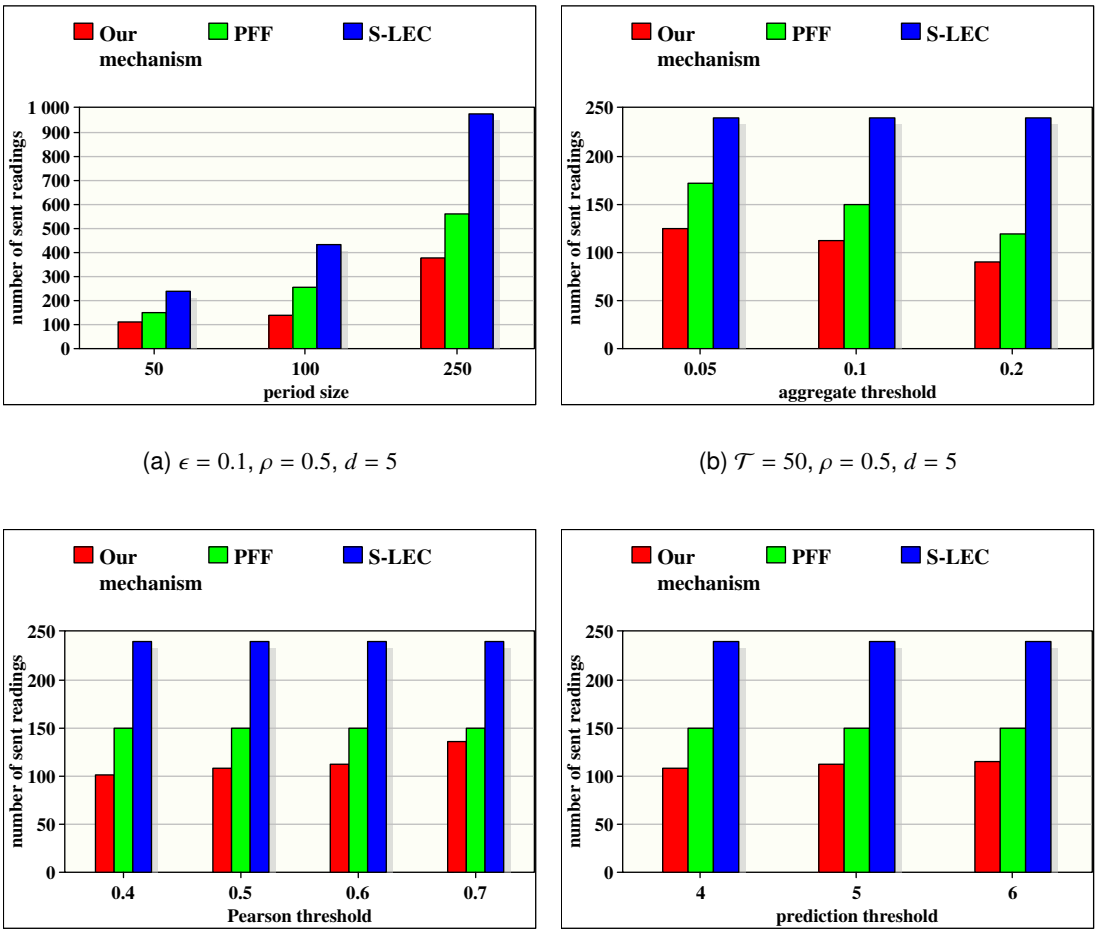
We observe that our mechanism can reduce the data transmission to the CH more than the PFF and S-LEC in all cases. Subsequently, it allows each sensor to send $9\%$ to $45\%$ of data less than PFF and $28\%$ to $67\%$ of data less than S-LEC. Furthermore, the obtained results show that: 1) the data transmission from the sensor, using our mechanism, increases with the increasing values of the period size (Fig. 4.5(a)) and the compression threshold (Fig. 4.5(c)). This is because, from one hand, the variance among the data calculated using ANOVA increases when the period size increases and, from the other hand, the collected readings become less redundant when the compression threshold increases. 2) The sensor sends, using our mechanism, less data to the CH when the aggregated threshold increases (Fig. 4.5(b)). This is due to the similarity among the collected readings, which increases with the increasing of the aggregate threshold. 3) The data transmission will not be highly affected when varying the predicted polynomial degree.

### 4.6.4/ Energy Consumption in Sensor

As previously mentioned, the energy consumed in the sensor node is highly related to the amount of its transmitted data. Figure 4.6 shows the remaining energy of temperature, humidity and light sensors in function of the period progress. In our simulations, we implemented the Heinzelman model proposed in [149] as one of the most models used to evaluate the energy consumption in sensor networks. Accordingly to this model, the energy consumption highly depends on the transmission and receiving operations while negligating the other factors (sensing and processing). Thus, the energy consumption of a sensor for transmitting its set of data $R_{i1}^{'p}$ with size $|R_{i1}^{'p}|$ to the CH located at distance $dist$ is:

$$E_{TX} = E_{elec} \times |R_{i1}^{'p}| \times 64 + \beta_{amp} \times |R_{i1}^{'p}| \times 64 \times dist^2 \tag{4.8}$$

where $64$ indicates the bit representation of each value, and $E_{elec}$ is the energy consumption of a sensor in its electronic circuitry (usually $E_{elec} = 50\,nJ/bit$), and $\beta_{amp}$ represents the




(a) $\epsilon = 0.1$, $\rho = 0.5$, $d = 5$

(b) $\mathcal{T} = 50$, $\rho = 0.5$, $d = 5$

(c) $\mathcal{T} = 50$, $\epsilon = 0.1$, $d = 5$

(d) $\mathcal{T} = 50$, $\epsilon = 0.1$, $\rho = 0.5$

Figure 4.5: Number of readings sent from each sensor to the CH.

energy consumption in RF amplifiers to compensate the loss (usually $\beta_{amp} = 100\, pJ/bit$).

Obviously, the remaining energy in each sensor proportionally decreases depending on the amount of data transmitted, with the progress of the period number. Subsequently, more the amount of data is reduced at each period, e.g. using on-period, and more the sensing frequency of the sensor is minimized at each round then less the available energy will be depleted. This supports the extension of the light sensor lifetime compared to those of other sensors due to the high redundancy level existing among light readings.

### 4.6.5/ PACKET TYPES STUDY AT CH

In Figure 4.7, we study the types of packets ($Off_Packet$, $Aggregate_Packet$, $Compressed_Packet$ and $Predicted_Packet$) received by the CH at each period. The obtained results show that the number of packets for each type can differ from one period to another for the same sensor (e.g. temperature, humidity or light) or they can differ for the different sensors at the same period. We can also observe that most of the received pack-

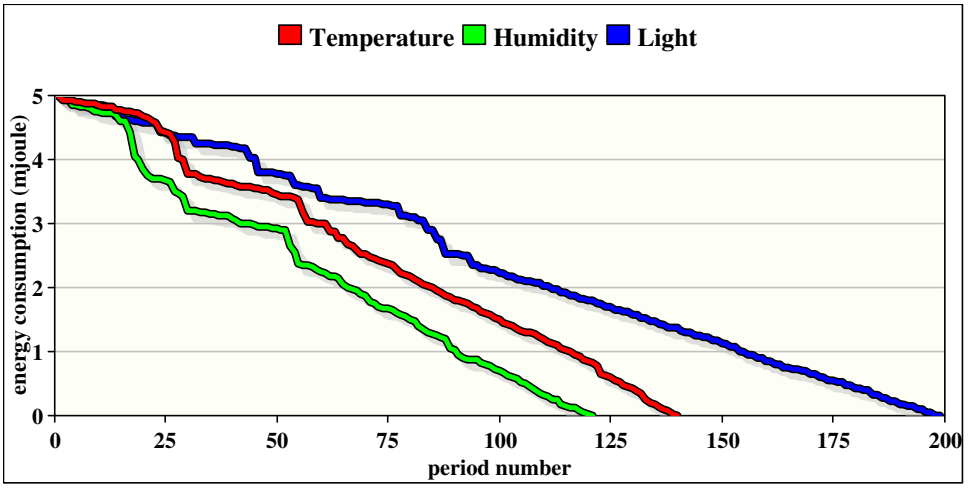


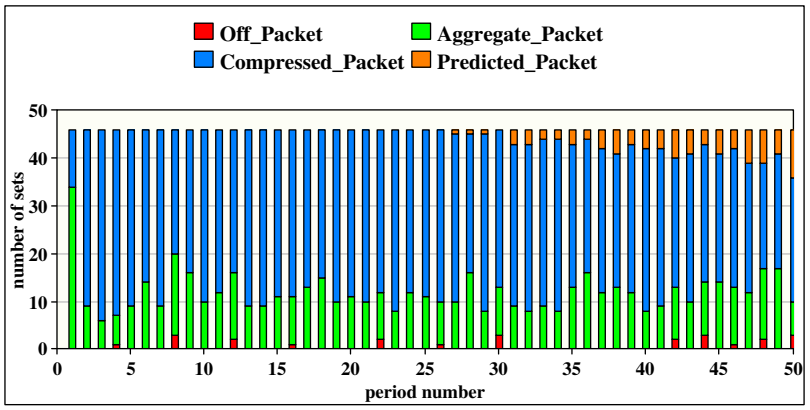

Figure 4.6: Remaining energy in a sensor in function of the period progress, $\mathcal{T} = 50$, $\epsilon = 0.1$, $\rho = 0.5$, $d = 5$.

ets are of type $Compressed_Packet$ followed by the $Aggregate_Packet$, $Predicted_Packet$ and $Off_Packet$ respectively, for various kind of sensors and for all periods. This is because compression is a compromised decision between aggregation and prediction approaches for energy saving and data accuracy at the same time. Furthermore, the results shows that the CH is receiving more packets of type $Predicted_Packet$ starting from the period number $27$ for the temperature and humidity readings, and from the period number $36$ for the light readings; this indicated that the energy of the sensors becomes low starting from such periods and the sensors have to reduce their data transmission in order to conserve their power supply. Finally, we observe that some sensors are delivering $Off_Packet$ to the CH indicating that the readings collected in successive periods are similar.

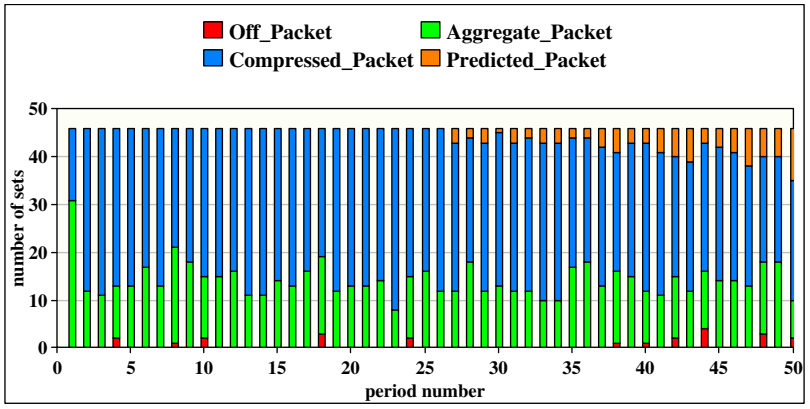
### 4.6.6/ In-Node Decision Study

Figure 4.8 shows the number of sets periodically sent from the CH to the sink after applying the in-node reduction algorithm (Algorithm 4.6). In addition to the $Off_Packet$ and $Predicted_Packet$, the CH sends a subset of the $Aggregate_Packet$, after removing the redundancy among them (Algorithm 4.4), and a subset of the compressed packets, after making them in clusters (Algorithm 4.5), to the sink. Thus, the obtained results are dependent on the period size ($\mathcal{T}$), the aggregation threshold ($\epsilon$) and the number of clusters ($K$) (Figure 4.8(a) to 4.8(c)) while they are not affected by the changing of the predicted polynomial degree (Figure 4.8(d)). Subsequently, we observe, using our mechanism, that the periodic number of sent sets decreases when the values of $\mathcal{T}$ or $K$ decrease, or the value of $\epsilon$ increases. This is because when $\mathcal{T}$ decreases or $\epsilon$ increases the similarity among the sensor sets will increase thus the CH will send less sets to the sink in order to avoid sending redundant data sets. Whilst, the decreasing of the cluster number leads to decrease the number of cluster centroids send to the sink. Furthermore, we observe that our mechanism outperforms PFF from $20\%$ to $40\%$ and S-LEC from $56\%$ to $73\%$ in terms of reducing the number of packets sent to the sink.

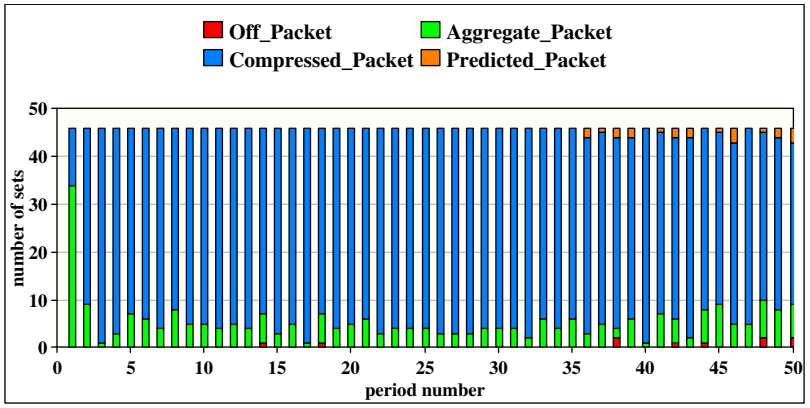
In Figure 4.9, we show an illustrative example of the packet types received by the CH during a period and after applying K-means over the $Compressed_Packet$. During




(a) temperature



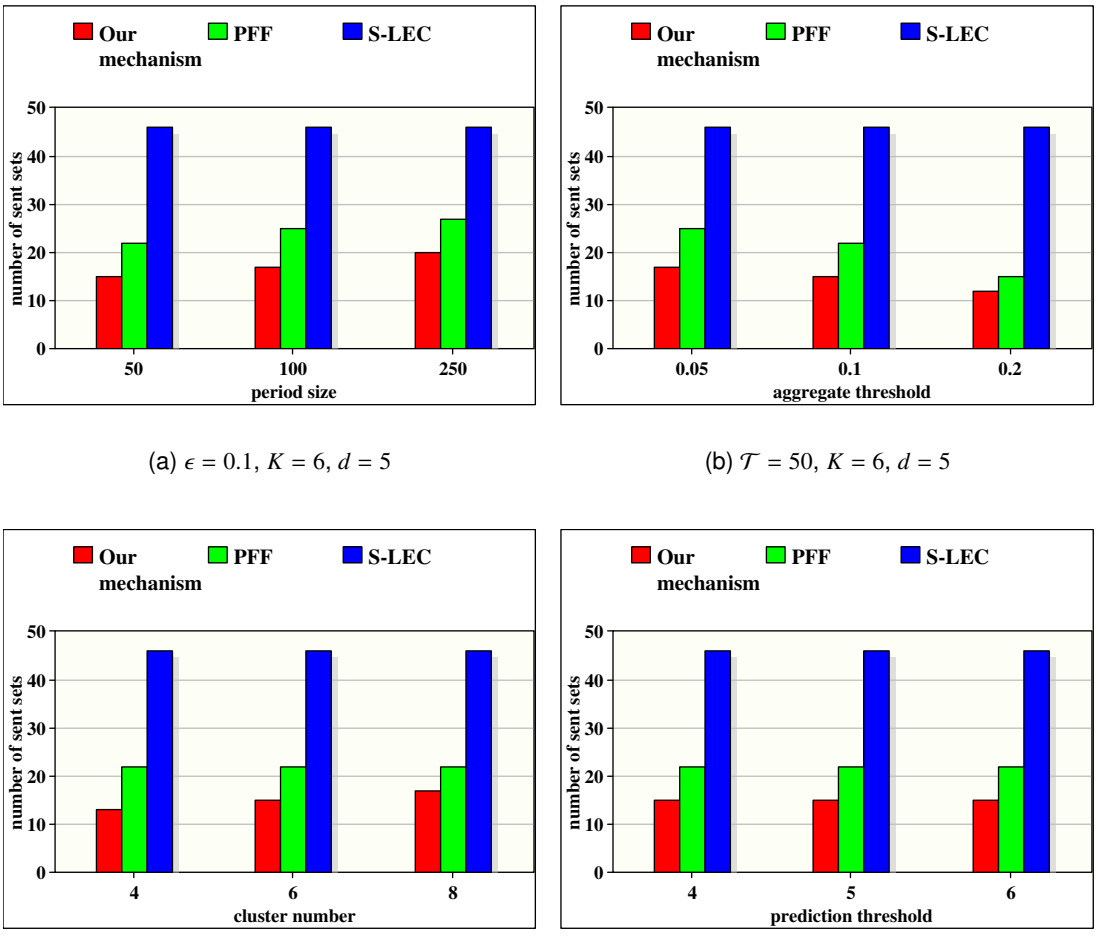

(b) humidity




(c) light

Figure 4.7: Variation of periodic packet types received by the CH.

this period, we observe that the CH receives 2 packets of type $Off_Packet$, 3 packets of type $Predicted_Packet$, 15 packets of type $Aggregate_Packet$ and 26 packets of type




(a) $\epsilon = 0.1$, $K = 6$, $d = 5$

(b) $\mathcal{T} = 50$, $K = 6$, $d = 5$

(c) $\mathcal{T} = 50$, $\epsilon = 0.1$, $d = 5$

(d) $\mathcal{T} = 50$, $\epsilon = 0.1$, $K = 6$

Figure 4.8: Number of sets sent periodically from the CH to the sink.

*Compressed_Packet*. Thus, after dividing the *Compressed_Packet* into 4 clusters, the following observations are eminent: 1) the sets are unequally distributed to the clusters; this is due to the random selection of the cluster centroids and the convergence function used in K-means. 2) The sensors in the same cluster are not necessary spatially correlated. 3) The temporal correlation among sensors can happen even they are not spatially correlated.

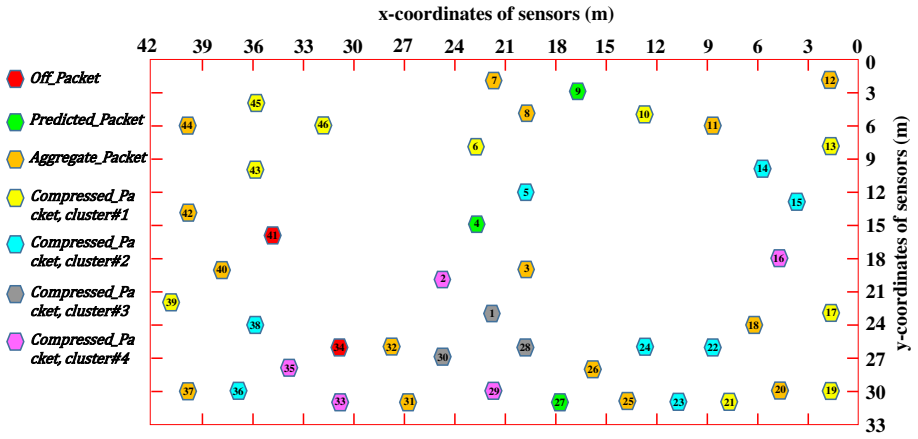



Figure 4.9: Illustrative example of packet types received by the CH during a period and after applying K-means over the compressed packets, $K = 4$.

## 4.7/ CONCLUSION

Data reduction will remain one of the main concerns for researchers in order to extend the sensing-based applications and deliver a useful data for the end user. In this chapter, we proposed a hybrid-based data collection mechanism, called All-in-One, with the aim to reduce the data transmission at several stages in the network. The proposed mechanism removes the redundancy existing among the collected data on on-period, in-period and in-node levels. Furthermore, on each level, we introduced several data reduction techniques while proposing hybrid-based approaches in order to optimize several performance metrics of the network. We conducted extensive simulations on real sensor data in order to evaluate the efficiency of our mechanism compared to other exiting techniques.

# 5

# AGGREGATION-SCHEDULING BASED MECHANISM FOR ENERGY-EFFICIENT MULTIVARIATE SENSOR NETWORKS

## 5.1/ INTRODUCTION

Nowadays, we live in the era of sensors in which a huge number of devices are connecting to the internet in order to monitor our surrounding and enhance the quality of life. From one hand, industries are continuously manufacturing new types of sensors for various domains in order to enable new services and applications that make life smarter and safety. According to a recent report provided by statista [156], the number of connected sensors are exponentially increasing every year starting from the beginning of the third millennium; $8$ billion sensor devices are offered in $2010$ and it is estimated to reach $30.9$ billion in $2026$. Such sensors take several forms such electrical, magnetic, thermal, acoustic, optical, chemical, etc. and they are integrated into a wide range of smart systems, such as: cities, buildings, transportation, healthcare, retail, industry etc [157]. On the other hand, researchers focus their works on several challenges related to sensor software rather than the hardware explored by industries. Indeed, they aim at proposing algorithms, protocols and techniques to overcome software challenges including the data collection and dissemination, energy conservation, network scalability, zone coverage, etc.

In this chapter, we are interested in the energy conservation as one of the most important issues studied in sensor applications. Unfortunately, sensor nodes are mostly equipped with limited battery power that cannot be recharged or costly to be replaced especially in harsh, far or dangerous zones. In the literature, researchers have mainly focused on two approaches to save the sensor battery and ensure a long network lifetime: data aggregation and scheduling strategies. The first approach aims to search the similarities among data collected by each sensor node and to only send useful information, e.g. non-redundant, to the sink. This will lead to reduce the amount of local (e.g. at node level) and global (e.g. at network level) data transmitted, enhance the power consumption, and prolong the network lifetime. The second approach aims to find the correlated nodes, whether temporally or spatially, then to select a subset of those having strong correlation to be in active mode while switching-off the others into a sleep mode. Consequently, scheduling approach will minimize the node activity as well as the network congestion and overload. Therefore, it becomes essential to design hybrid solutions that take advan-

tages from both aggregation and scheduling approach in order to increase the network performance and enhance its resources.

In this chapter, we propose a hybrid mechanism called AGING that combines aggregation and scheduling for energy-efficient multivariate sensor networks. Subsequently, we assume that each node $N_i$ contains a set of $Q$ sensors and it collects, during each period $p$, a matrix of data $M_i^p$ to be sent to the CH at the end of the period (cf. section 1.5.2 in Chapter 1). AGING divides the network into clusters where data are sent periodically from the nodes to their cluster-heads (CHs). Then, AGING proposes a data aggregation phase at the node level to minimize the data transmission ratio between the nodes and the CHs, based on a user-defined score table and a multi-aggregation mechanism. Once the data are received by the CHs, AGING introduces a scheduling strategy that switches nodes having high spatial-temporal correlations into sleep/active modes. At this level, the correlation between nodes are represented by a graph followed by a coloring-map algorithm and a scheduling strategy to select the set of actives nodes in the next periods. Therefore, our mechanism shows its high performance in terms of saving the node energies, eliminating the redundant data and reducing the congestion of the sensor network.

The remainder of this chapter is organized as follows: Section 5.2 describes our mechanism while detailing each of its phases. Section 5.3 describes the implementation of our mechanism and explains the obtained results. Finally, Section 5.4 concludes the chapter.

## 5.2/ AGING MECHANISM

Indeed, the number of proposed works in aggregation and scheduling in sensor networks is increasing and they provide efficient solutions for energy conservation, however they mostly suffer from several disadvantages. First, most of techniques take advantages from either aggregation or scheduling approaches to reduce the data transmission in the network but not from both ones. This is mainly because of avoiding the trade-off between the energy conservation and the accuracy that may highly affected in case of a combined approach is adapted. Second, researchers mostly applied their works on one network side (either nodes, CHs or sink), but not at several ones simultaneously. Third, some techniques are very complex and not suitable for limited-resources (memory and CPU) sensors. In this chapter, we present a less-complex mechanism called AGING that allows to save the node energies while ensuring a high level of data accuracy. Mainly, AGING relies on multi-aggregation algorithm at the sensor nodes as well as a coloring-map algorithm and scheduling strategy at the CHs. Figure 5.1 shows the architecture of AGING mechanism with the two main phases and various used algorithms that will be detailed in next sections.

### 5.2.1/ AGGREGATION PHASE

Indeed, the multivariate data collected in sensor networks are mostly of huge size due to the periodic collection model adapted in most applications. Such amount of data will highly affect the network performance in terms of communication complexity, energy consumption, overload, and congestion [158]. Furthermore, such data are redundant due to the spatial-temporal correlation existing in most applications. Therefore, in order to overcome these challenges, the amount of periodic data sent by each node to the CH

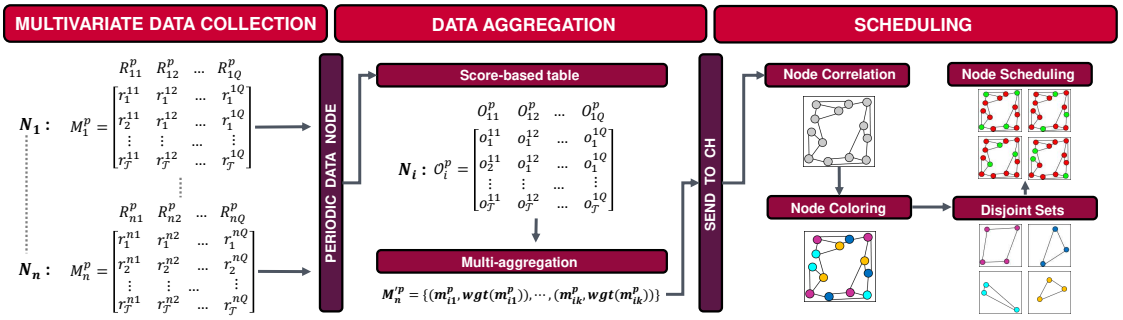



Figure 5.1: Architecture of AGING mechanism.

should be locally reduced, which can mostly done by eliminating the redundancies existing among the collected readings. The first phase in our mechanism, e.g. aggregation phase, aims to reduce the size of reading matrix $M_i^p$ collected by each node before sending it toward the CH. Subsequently, this phase works on two steps: first, it defines a score-based table to determine the similarities between the data collected by each sensor then it proposes a multi-aggregation technique for data transmission reduction between node and CH.

#### 5.2.1.1/ SCORE-BASED TABLE

The readings collected by each sensor implemented on a node are highly dependent on the variation of the monitored condition. Thus, when the condition is varying slowly, the redundancy among data collected by the corresponding sensor will increase, and vice versa. Hence, in order to determine the redundancy among data collected by each sensor, we define a score table based on the criticality of the monitored data and the monitored condition. Subsequently, we use the same score table defined in the third chapter and shown in Figure 3.3, where a normal range, $]r_l^j, r_u^j[$, is set for each condition $j$ monitored by the sensor $S_{ij}$ in $N_i$. Furthermore, readings outside of this range indicate an abnormal condition situation with a score ranging in $[0,3]$ and a set of thresholds is defined for all sensors in a node as follows: $\mathcal{H} = \{\delta_1, \delta_2, \ldots, \delta_K\}$. According to the score table, each node will calculate the score $o_t^{ij}$ for each reading $r_t^{ij}$ then forms the score matrix $O_i^p$ of $M_i^p$ at the end of each period as follows:

$$O_i^p = \begin{array}{c} \\ t_1 \\ t_2 \\ \vdots \\ t_{\mathcal{T}} \end{array} \begin{array}{c} \begin{array}{cccc} O_{i1}^p & O_{i2}^p & \ldots & O_{iQ}^p \end{array} \\ \begin{pmatrix} o_1^{i1} & o_1^{i2} & \ldots & o_1^{iQ} \\ o_2^{i1} & o_2^{i2} & \ldots & o_2^{iQ} \\ \vdots & \vdots & \ldots & \vdots \\ o_{\mathcal{T}}^{i1} & o_{\mathcal{T}}^{i2} & \ldots & o_{\mathcal{T}}^{iQ} \end{pmatrix} \end{array} \begin{array}{l} \\ = O_{i1}^p \\ = O_{i2}^p \\ \vdots \\ = O_{i\mathcal{T}}^p \end{array} \tag{5.1}$$

#### 5.2.1.2/ MULTI-AGGREGATION TECHNIQUE

After determining the score matrix of periodic data collected by each sensor, we propose a multi-aggregation technique to efficiently reduce the size of $M_i^p$ before sending it to the appropriate CH. The idea behind our technique is to search the similarities among data collected in successive slots then to only send those with different scores with the

previous readings. Subsequently, the accuracy of the sent information can be maintained by defining a weight variable $wgt$ that indicates the number of successive reading sets with the same score. Algorithm 5.1 shows the process of the multi-aggregation technique applied periodically at each node. The algorithm takes the matrix of readings collected during a period and returns a reduced list of reading sets, e.g. $M_i^{'p}$, to send to the CH at the end of the period. The process starts by adding the first reading set collected in the first slot with a weight of 1 to the set of final reading list (lines 1-2). Then, for the next slots, the reading set is added to the final reading list only if they have different score from the last reading list inserted in the list (lines 4-6). Otherwise, the monitored zone is considered in a stable status and they are removed from $M_i^p$ while adding the weight of the last reading set inserted in $M_i^{'p}$ by 1 (line 8).

**Algorithm 5.1** Multi-Aggregation Algorithm.

**Require:** Node: $N_i$; Set of sensors: $N_i = \{S_{i1}, S_{i2}, \ldots, S_{iQ}\}$; Period: $p$; Matrix of readings: $M_i^p$.
**Ensure:** List of reading sets: $M_i^{'p}$.
1: $O_i^p$ = calculate the score matrix of $M_i^p$
2: $M_i^{'p} \leftarrow \{(m_{i1}^p, 1)\}$
3: **for** each reading set $m_{it}^p \in M_i^p$ where $t \geq 2$ **do**
4: assume $m_{ik}^p$ is the last reading set inserted in $M_i^{'p}$
5: **if** $O_{it}^p = O_{ik}^p$ **then**
6: $wgt(m_{i1}^p) = wgt(m_{i1}^p) + 1$
7: **else**
8: $M_i^{'p} \leftarrow M_i^{'p} \cup \{(m_{it}^p, 1)\}$
9: **end if**
10: **end for**
11: **return** $M_i^{'p}$

### 5.2.2/ SCHEDULING PHASE

At the end of each period, the CH will receive the reduced reading sets sent by the sensors, e.g. $M^{'p} = \{M_1^{'p}, M_2^{'p}, \ldots, M_n^{'p}\}$ where $M_i^{'p}$ is the reading set sent from the node $N_i$. Indeed, $M^{'p}$ contains a high redundancy level due to the spatial-temporal correlations existing among the nodes. Hence, it comes the role of CH to eliminate the redundancy and reduce the amount of data collected in sensor networks. This is done by using a scheduling approach in which a set of non-correlated nodes is selected to collect the data instead of the whole one. The new scheduling strategy proposed in this phase is composed of the following steps:

#### 5.2.2.1/ GRAPH-BASED SPATIO-TEMPORAL NODE CORRELATION

in sensor networks, two nodes are considered correlated if they are geographically close and generate similar data. From one hand, the spatial correlation occurs when the sensing range, e.g. $S_r$, of two nodes $N_i$ and $N_j$ are overlapped. This happens when the

geographical Euclidean distance, $E_g$, between them is less than a defined geographical threshold $G$ as follows:

$$E_g(N_i, N_j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \leq G \tag{5.2}$$

where $(x_i, y_i)$ is the geographical position of the node $N_i$, and $G \in [0, 2 \times S_r]$. On the other hand, the temporal correlation occurs when a high similarity, or a low dissimilarity, among two matrices data $M_i^{'p}$ and $M_j^{'p}$ collected by $N_i$ and $N_j$ is noticed. Subsequently, we also used the Euclidean distance in order to calculate the dissimilarity among data matrices $M_i^{'p}$ and $M_j^{'p}$ of both nodes $N_i$ and $N_j$, after desaggregating them into raw data, as follows:

$$E_d(M_i^{'p}, M_j^{'p}) = \sqrt{\sum_{q=1}^{Q} \sum_{t=1}^{\mathcal{T}} (r_t^{iq} - r_t^{jq})^2} \tag{5.3}$$

where $E_d$ indicates the temporal distance between both data matrices, $r_t^{iq} \in M_i^{'p}$ and $r_t^{jq} \in M_j^{'p}$. However, instead of using the threshold-based approach, we adapt the K-nearest neighbor (KNN) algorithm in order to determine the best temporal correlated nodes for a given one. Thus, by assigning a value to $K$, we select the $K$ nodes having the minimum Euclidean distance ($E_d$) to the data collected by a node $N_i$ as the temporal correlated nodes to such node. Accordingly, less the value of $K$ is, more the nodes are considered temporally correlated, and vice versa. Therefore, two nodes $N_i$ and $N_j$ are considered spatially-temporally correlated if they meet equations 5.2 and 5.3 (with KNN condition) at the same time.

#### 5.2.2.2/ Graph Construction and Nodes-Coloring

once all the correlations among nodes are found, the CH considers the network as an undirected graph $G(V, E)$ where the set of vertices $V$ indicates the nodes and the set of edges $E$ represents the connections between the nodes. Subsequently, two nodes are connected on the graph if they are spatially-temporally correlated. After that, we color connected vertices with different colors in order to minimize the correlations among network nodes and remove the redundancies among the collected data. Indeed, there are many colored-based graph algorithms used in the literature [159]. However, we focus on the Backtracking as one of the most well-studied and used algorithms. Historically, Backtracking has been used in a wide range of applications, such as game players and power systems, and been integrated into several sciences such as mathematics, economics, and data analysis [160]. In our case, the graph vertices are first numbered according to the ids of the nodes, then the backtracking algorithm is recursively applied to build a solution incrementally, one color at a time, removing those solutions that fail to satisfy the constraint of obtaining different colors of connected vertices at any point of time [161]. After coloring all vertices, we obtain a $(K + 1)$-coloring graph, where none of the nodes has similar color to its correlated one.

#### 5.2.2.3/ Disjoint Sets and Node Scheduling

Let consider a list $\mathcal{L}$ of $(K+2)$ node sets formed after the node-coloring step as follows: $\mathcal{L} = \{L_1, L_2, \ldots, L_{K+2}\}$. The first node set $L_1$ consists of all nodes in the cluster while each of the other sets $L_i$ consists of $|L_i|$ nodes having the same color. Then, our scheduling strategy operates in rounds, where each round $\pi$ is composed of $(K+2)$ successive periods. In the first period, all nodes are in active mode and will collect data and send them to its CH. Whilst, in the next periods of the round, we only activate the nodes in one disjoint set while switching other ones to the sleep mode. Therefore, this strategy will allow to update the disjoint node sets based on the new spatio-temporal correlation between nodes in the first period and, accordingly, to avoid collecting similar data by neighboring nodes in the remaining periods.

Figure 5.2 shows an illustration example of the node correlation and scheduling process of the second phase in our mechanism. We consider a CH with a set of $13$ nodes where the value of $K$ is fixed at $3$. After receiving data from all nodes at the first period, the CH finds the graph node correlation followed by applying the $4$-coloring graph algorithm, then it determines the $4$ disjoint sets, e.g. $L_2$ to $L_5$, using the Backtracking algorithm. Finally, the round is composed of $5$ periods where nodes of each set are activated in the corresponding period of the round. For instance, the set of nodes $L_2 = \{1, 4, 8, 10\}$ will collect the data in the second period while switching-off the others to sleep mode.

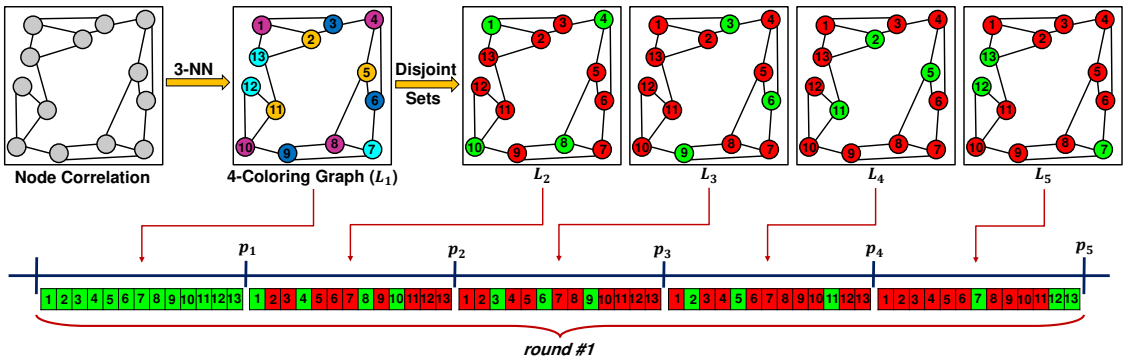



Figure 5.2: Illustrative example of node correlation and scheduling.

## 5.3/ Performance Evaluation

In our simulations, we used the real data collected and available online by the Intel Berkeley Research Lab [145]. Mainly, the laboratory deploys $46$ nodes of type Mica2Dot with weather boards where each node consists of $3$ sensors collecting temperature, humidity, and light readings. The dataset contains about $2.3$ million readings collected during about $40$ days with a sensor sampling fixed to one reading per $31$ seconds. In our simulations, we assumed a common CH located at the center of the lab where all sensors read their corresponding data from their corresponding files and send them periodically to this CH. We compared our results to those of PPMC [130] and SFDC [129] respectively. The parameter used in our simulations are shown in Table 5.1.

| Parameter | Description | Values |
|---|---|---|
| $\mathcal{T}$ | period size | 200, 500, 1000 |
| $\delta_j$ | aggregation threshold | 0.01, 0.05, 0.1 |
| $S_r$ | sensing range | 10 meters |
| $G$ | geographical threshold | $S_r$, $\frac{3}{2} \times S_r$ |
| $K$ | temporal threshold | 3, 5, 7 |
| $E_i$ | initial node energy | 10 mJ |

Table 5.1: Simulation environment.

### 5.3.1/ Data Transmission Ratio at Node Level

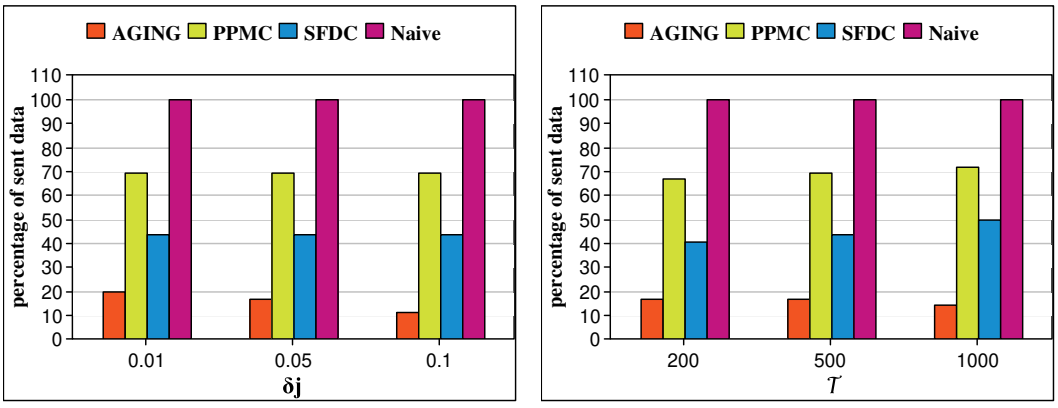
In Figure 5.3, we show the average percentage of readings sent from each sensor to the CH at each period. The obtained results are highly dependent on the aggregation threshold defined in the score table and the period size respectively. We show that AGING outperforms PPMC and SFDC in terms of reducing the data transmission ratio from the nodes in all cases; naïve indicates data transmission without applying local approach at the node level. Subsequently, AGING reduces up to $83\%$, $73.3\%$ and $88.4\%$ of data transmission ratio compared to PPMC, SFDC and naïve approaches. Furthermore, the results of AGING show the following observations:

- The percentage of transmitted data with AGING decreases with the increasing of the aggregation threshold. This is because the successive readings having the same score will increase when the value of $\delta_j$ increases thus, the multi-aggregation mechanism used in AGING will eliminate more redundant data. For instance, the percentage of sent data reduces from $19.8\%$ to $11.6\%$ when $\delta_j$ increases from 0.01 to 0.1.

- The percentage of data transmission with AGING decreases with the increasing of the period size. This is because the similarity among the collected data increases when increasing the value of $\mathcal{T}$, which consequently reduces the variation of readings scores and the corresponding data transmission.

### 5.3.2/ Average Node Lifetime

In this section, we study the performance of the proposed mechanism in terms of saving the node energies and extending the network lifetime. Figure 5.4 shows the number of periods in which a node is operational in function of the values of $\delta_j$ and $\mathcal{T}$ used in the aggregation phase (Figures 5.4(a) and 5.4(b) respectively) and those of $G$ and $K$ used in the scheduling phase (Figures 5.4(c) and 5.4(d) respectively). The obtained results show that AGING mechanism allows each node to significantly saving its energy and extend its lifetime compared to other approaches. This confirms the behavior of AGING in terms of eliminating the data redundancies at the node level and removing the node correlation and reducing the node activity at the CH level. Furthermore, the following observations are eminent:



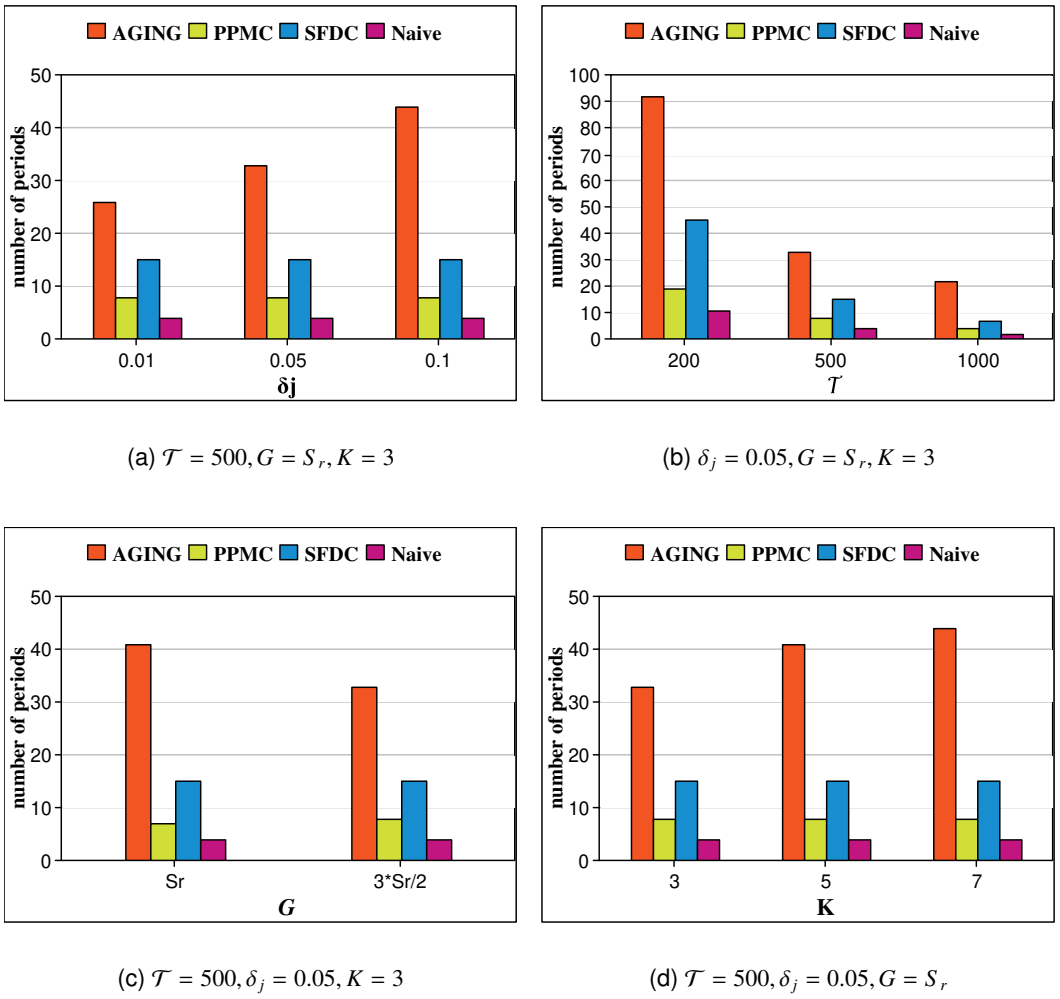

(a) $\mathcal{T} = 500$ (b) $\delta_j = 0.05$

Figure 5.3: Average percentage of data sent from each node to the CH at each period.

- AGING increases the node lifetime up to $6$, $3$ and $11$ times compared to PPMC, SFDC and naïve techinques.
- The node lifetime is further improved using AGING when the aggregation threshold increases (Figure 5.4(a)) or the period size decreases (Figure 5.4(b)). This is due to the amount of data transmitted from each node when $\delta_j$ increases or $\mathcal{T}$ decreases. For instance, the node lifetime is increased by $69.2\%$ when $\delta_j$ is changed from $0.01$ to $0.1$, and by $318.5\%$ when $\mathcal{T}$ is varied from $1000$ to $200$.
- The node lifetime is significantly extended with the decreasing of the geographical threshold (Figure 5.4(a)). This is because the spatial correlation between nodes will increase when decreasing the corresponding spatial threshold. For instance, we see that the node lifetime is improved by $19.5\%$ when the value of $G$ is varied from $S_r$ to $3 \times S_r/2$.
- AGING is further extending the node lifetime when the increasing value of temporal threshold (Figure 5.4(d)). This is because that more nodes will be considered as temporally correlated when $K$ increases, thus the scheduling phase will switch more nodes to the sleep mode and consequently increasing its lifetime.

### 5.3.3/ Percentage of Data Loss

Indeed, performing node scheduling without taking into account the data accuracy is not an efficient way for sensor networks although it conserves the node energies. In this section,we study the accuracy of three techniques in terms of preserving the integrity of the information sent to the end user. Figure 5.5 shows the percentage of data loss for the three techniques at the end of the simulation while varying the values of the parameters similarly to those in Figure 5.4. In our simulations, a reading collected by a node is considered as loss reading if no similar one is sent by such node or its correlated ones to the end user. Thus, the data loss is highly related to the amount of data sent from




(a) $\mathcal{T} = 500, G = S_r, K = 3$

(b) $\delta_j = 0.05, G = S_r, K = 3$

(c) $\mathcal{T} = 500, \delta_j = 0.05, K = 3$

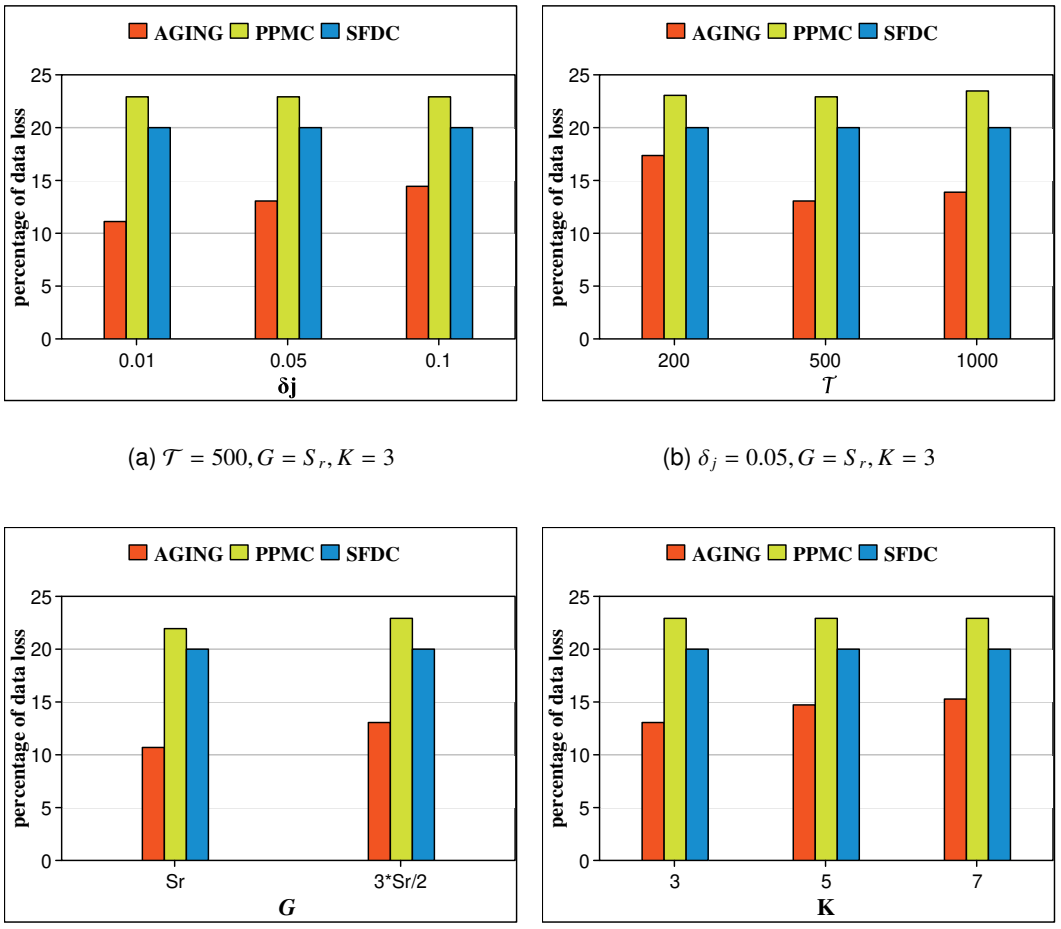
(d) $\mathcal{T} = 500, \delta_j = 0.05, G = S_r$

Figure 5.4: Average lifetime of each node.

the nodes where less sent data mostly leads to increase the data loss. Consequently, the obtained results in Figure 5.5 show three observations: first, the percentage of data loss in AGING is less than those in PPMC and SFDC. For instance, the data loss in AGING varies between $10.8\%$ and $17.5\%$ while it reaches up to $23.6\%$ and $20\%$ in PPMC and SFDC respectively. This is because the similarity condition used in the aggregation phase and the correlation condition used in the scheduling phase perform efficient data reduction for only the redundant ones. Second, the percentage of data loss in AGING decreases with the decrease of $\delta_j$ (Figure 5.5(a)) or the increase of $K$ (Figure 5.5(d)). Third, the percentage of data loss is almost fix when changing the period size (Figure 5.5(b)) and the geographical threshold (Figure 5.5(c)).

### 5.3.4/ Active Nodes vs Zone Coverage

In this section, we study the performance of the scheduling phase in terms of the number of active nodes and the coverage zone. Figure 5.6 shows the variation of the coverage zone area (right y-axis) in function of the number of active nodes (left y-axis) for




(a) $\mathcal{T} = 500, G = S_r, K = 3$

(b) $\delta_j = 0.05, G = S_r, K = 3$

(c) $\mathcal{T} = 500, \delta_j = 0.05, K = 3$

(d) $\mathcal{T} = 500, \delta_j = 0.05, G = S_r$

Figure 5.5: Percentage of data loss after applying scheduling strategies.

some fixed values of parameters. Indeed, three observations are eminent in the obtained results. First, the number of active nodes show that the nodes in the lab are highly correlated and their collected data look very similar; consequently, an average of $18$ nodes were active in each period. Second, the scheduling phase ensures a high level of coverage zone during the entire network lifetime; the coverage zone ratio varies between $70.6\%$ and $100\%$ for the first $28$ periods where most of the nodes were operational with sufficient battery level. Third, we notice that active nodes and coverage ratio metrics are highly and proportionally correlated where the increase of the number of operational nodes leads to increase the coverage area of the zone, and vice versa. Therefore, this confirms the behavior of the scheduling phase by reducing the number active nodes while ensuring a high coverage ratio for the monitored zone.

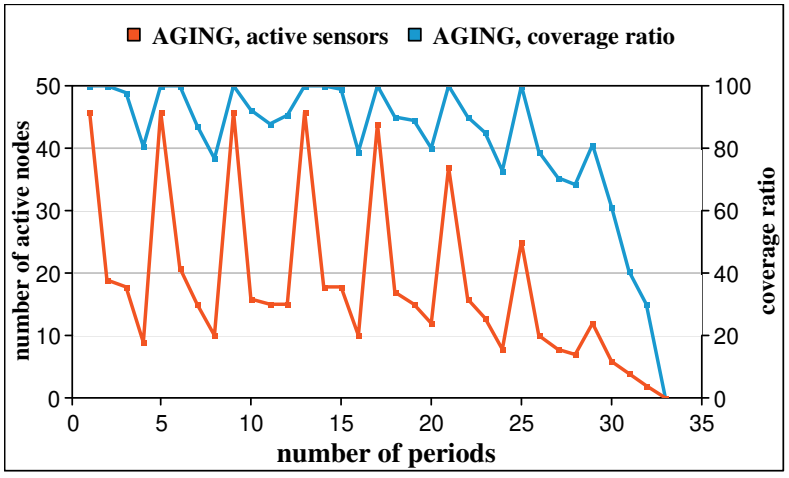



Figure 5.6: Variation of the coverage zone area in function of the number of active nodes, $\mathcal{T} = 500, \delta_j = 0.05, G = S_r, K = 3$.

## 5.4/ CONCLUSION

As the world becomes more smarter every year, the need of sensing technology will take more attention from industries and researchers. While the companies are trying to further investigate in the sensor hardware, researchers are targeting to design new techniques and mechanisms to overcome some software challenges, especially the energy conservation. In this chapter, we have proposed a hybrid mechanism called AGING that takes advantages from both aggregation and scheduling approaches for energy-efficient multivariate sensor networks. AGING is based on the clustering scheme and consisted of two main phases. The first phase is called data aggregation in which a user-defined score table and a multi-aggregation mechanism are used in order to reduce the amount of periodic data transmitted by each node. The second phase is called scheduling where we searched the node correlation at the CH then we switched those having strong correlation into sleep/active modes. We demonstrated the effectiveness of our mechanism based on real sensor data and in terms of energy saving, increasing network lifetime, and data accuracy.

# 6

# CONCLUSIONS AND PERSPECTIVES

## 6.1/ CONCLUSIONS

Due to potential applications, sensing-based technology is expecting to grow exponentially in the next years leading to a radical change in various domains. However, with the increasing number of sensing devices, the amount of generated and collected data will also increase. Thus, proposing new techniques and mechanisms that allow to keep useful data and remove redundant ones will become more crucial for the energy-conservation and decision-making in sensor technology.

In this thesis, we proposed energy-efficient and data reduction mechanisms based on cluster architecture and periodic model for resource-constrained sensor networks. The proposed mechanisms handled the huge amount of data collected in sensor networks thus, they offer efficient solutions to prolong the network lifetime and enable less-complex data analysis for decision makers. The main goal of our mechanisms is to remove the data redundancies existing at sensor nodes, e.g. on-period and in-period, and CHs, e.g. in-node, as well as to provide decision-making models at the sink.

At the sensor level, we proposed several data collection and transmission methods to reduce the sensing and communication operations. From on hand, we presented data prediction, aggregation and compression algorithms based respectively on Newton forward difference, divide-and-conquer and Pearson coefficient to eliminate on-period data redundancy at each period. On the other hand, we presented two algorithms based respectively on sampling frequency adaptation and on-off transmission to eliminate in-period data redundancy among data collected by each sensor in successive periods.

At the CH level, we proposed several data reduction techniques in order to remove in-node data redundancy between neighboring nodes. The first technique is based on data clustering and introduces a new version of K-means, called PK-means, that groups sensor nodes generating similar data into clusters for eliminating redundant ones. The second technique is a data fusion method that proposes a support-confidence algorithm to fuse similar sensor data before sending to the sink. The third technique is an in-network data aggregation that allows to CH to find data correlation between sensor nodes based on distance function. The last technique is a scheduling strategy to switch-off correlated neighboring nodes generating similar data into sleep/active modes.

At the sink level, we proposed a real-time decision-making model that may be customized depending on the context and circumstances of the monitored application. Our model is an expert-defined and it is based on two customizable tables, e.g. the score decision

and early decision table, that are used by the application services staff to determine the real-time status of the monitored zone.

## 6.2/ PERSPECTIVES

We have two directions of perspectives in order to enhance our work in this thesis: Short to Mid Term or Long Term. The first direction of perspectives are related to the mechanisms proposed in this work while the second direction of perspectives are open issues in energy-conservation and data handling for sensor networks.

### 6.2.1/ SHORT TO MID TERM PERSPECTIVES

In this section, we give some perspectives in order to improve or extend the proposed mechanisms at sensor nodes, CHs or sink presented in this work.

**1.** Many enhancements can be made on our mechanisms at the sensor nodes to enhance their performance:

- We seek to test another interpolation methods, such as Lagrange interpolation or Least Square Error, to compare for better results of prediction.
- We plan to use optimization techniques, such as genetic algorithm and particle swarm optimization, in order to dynamically find the optimal values of compression and aggregation thresholds. This will help the decision-makers to make a trade-off between data accuracy and energy-saving depending on the application requirements.
- We seek to take into consideration another information when adapting the sensor frequency. Example of information may include the “sensor position” that can help to prevent neighboring nodes to take readings at the same slots in the period.
- We plan to add a shift phase between successive transmissions of a sensor in order to reduce the congestion in the network and minimize the packet loss.

**2.** We have three enhancements to increase the performance of our mechanisms at the CH:

- An important direction to follow and study concerns the reduction of the complexity of the PK-means algorithm to further reduce its latency.
- We plan to add a prediction model at CH/sink so the sink can predict the data eliminated at CH and improve the accuracy of data fusion and in-network data aggregation.
- We seek to take into consideration more metrics such as node correlation, remaining node energy, and application criticality, when scheduling the sensor nodes in chapter 4. Particularly, we can focus on optimization algorithms in order to select the best active sensors that ensure a high coverage of the network.

3. We seek to enhance the decision-making model proposed at the sink in chapter 3. Particularly, we plan to focus on another decision systems such as decision tree, Markov chain or fuzzy set. Then, we plan to propose a dynamic model that allows to build the customizable tables in our mechanism according to the application requirements.

4. We plan to merge all the proposed mechanisms in one framework in order to better conserve the energy and ensure further quality-of-service for the decision-making process in sensor networks.

5. We seek to adapt our proposed mechanism to other network architecture such as tree-based or chain-based in order to evaluate their performance.

6. It is interesting to perform more general real experiments in order to evaluate the performance of all the proposed mechanisms in real world applications.

### 6.2.2/ LONG TERM PERSPECTIVES

Despite the great efforts made in data handling and energy-conservation in sensor networks, the area is still largely open to research. Consequently, more efforts should be investigated in several open research issues related to data handling, energy conservation and decision-making that are yet unexplored or, sometimes, need to be further explored. In this section, we would like to attract the attention of researchers to such issues in order to improve the performance of sensor networks.

The first issue to be more investigated is the data collection in sensor networks. With the rapid development in artificial intelligence (AI) techniques, it becomes important to integrate such techniques in sensor nodes in order to collect and transmit data in a more intelligent way. This will help to facilitate the decision-making process as well as saving the network energy.

The second issue that remains largely unexplored in sensor networks is the multimedia aspect. Unfortunately, most of the proposed techniques are dedicated to homogeneous sensor network, especially numerical data, while few ones are targeted sensor network with heterogeneous data such as numerical, images, video, etc. Indeed, multimedia sensor networks offer efficient solutions for many applications, especially transportation and military systems, and help in increasing the reliability of the collected data thus, the decision-making. In such type of networks, energy saving becomes more crucial due to the huge amount of collected and transmitted data thus, proposing new data reduction techniques for multimedia sensor networks is becoming essential to be focused.

Another important issue that did not yet largely explored is the decision-making in sensor networks. Nowadays, with the emergence of distributed systems (such as Hadoop and Spark) and the cloud computing, Big data analytics offer an efficient solution to propose new decision-making models in sensor networks. Subsequently, such systems can help in storing the huge amount of data collected in sensor network, especially the multimedia ones, and, from other hand, to provide a fast data processing and real-tie decision-making for decision makers.

Finally, time synchronization is a significant challenge in periodic sensor network which is not largely focused by researchers. Since data should be sent periodically, any loss or delayed can change the data time synchronization at the sink which raises a problem in

decision making. Therefore, more techniques need to be proposed in order to guarantee an accurate time information for the collected data in periodic sensor networks.

# PUBLICATIONS

## ACCEPTED AND SUBMITTED JOURNALS

[**1**] Marwa Ibrahim, Hassan Harb, Ali Mansour, Abbass Nasser and Christophe Osswald. All-in-one: Toward hybrid data collection and energy saving mechanism in sensing-based IoT applications. *Peer-to-Peer Networking and Applications Journal*, Springer Publisher, Vol. 14, pages 1154–1173, 2021.

[**2**] Marwa Ibrahim, Hassan Harb, Ali Mansour, Abbass Nasser and Christophe Osswald. A Scheduling-Based Strategy for Energy-Efficient Multivariate Sensor Networks. *Submitted to IEEE Sensors Journal*, September 2021.

## ACCEPTED AND PUBLISHED CONFERENCES

[**1**] Marwa Ibrahim, Hassan Harb, Abbass Nasser, Ali Mansour and Christophe Osswald. ON-IN: An On-Node and In-Node Based Mechanism for Big Data Collection in Large-Scale Sensor Networks. *27th European Signal Processing Conference (EUSIPCO)*, Coruna, Spain, 2-6 Sept, pages 1–5, 2019.

[**2**] Marwa Ibrahim, Hassan Harb, Abbass Nasser, Ali Mansour and Christophe Osswald. Adaptive Strategy and Decision Making Model for Sensing-Based Network Applications. *19th International Symposium on Communications and Information Technologies (ISCIT)*, Ho Chi Minh City, Vietnam, 25-27 Sept, pages 96–101, 2019.

# BIBLIOGRAPHY


**[1]** Dionisis Kandris, Christos Nakas, Dimitrios Vomvas, and Grigorios Koulouras. Applications of wireless sensor networks: an up-to-date survey. *Applied System Innovation*, 3(1):1–24, 2020.

**[2]** Parvaneh Asghari, Amir Masoud Rahmani, and Hamid Haj Seyyed Javadi. Internet of things applications: A systematic review. *Computer Networks*, 148:241–261, 2019.

**[3]** Alireza Souri and Monire Norouzi. A state-of-the-art survey on formal verification of the internet of things applications. *Journal of Service Science Research*, 11(1):47–67, 2019.

**[4]** Wazir Zada Khan, MH Rehman, Hussein Mohammed Zangoti, Muhammad Khalil Afzal, Nasrullah Armi, and Khaled Salah. Industrial internet of things: Recent advances, enabling technologies and open challenges. *Computers & Electrical Engineering*, 81:1–13, 2020.

**[5]** Mohammad Abdul Matin and MM Islam. Overview of wireless sensor network. *Wireless sensor networks-technology and protocols*, pages 1–3, 2012.

**[6]** L Minh Dang, Md Piran, Dongil Han, Kyungbok Min, Hyeonjoon Moon, et al. A survey on internet of things and cloud computing for healthcare. *Electronics*, 8(7):768, 2019.

**[7]** Cristina Paniagua and Jerker Delsing. Industrial frameworks for internet of things: A survey. *IEEE Systems Journal*, 15(1):1149–1159, 2020.

**[8]** Divyansh Thakur, Yugal Kumar, Arvind Kumar, and Pradeep Kumar Singh. Applicability of wireless sensor networks in precision agriculture: A review. *Wireless Personal Communications*, 107(1):471–512, 2019.

**[9]** Muhammad Ayaz, Mohammad Ammad-Uddin, Zubair Sharif, Ali Mansour, and El-Hadi M Aggoune. Internet-of-things (iot)-based smart agriculture: Toward making the fields talk. *IEEE Access*, 7:129551–129583, 2019.

**[10]** Muhammad Muzzammil, Niaz Ahmed, Gang Qiao, Imran Ullah, and Lei Wan. Fundamentals and advancements of magnetic-field communication for underwater wireless sensor networks. *IEEE Transactions on Antennas and Propagation*, 68(11):7555–7570, 2020.

**[11]** Ruhul Khalil, Mohammad Babar, Tariqullah Jan, and Nasir Saeed. Towards the internet of underwater things: Recent developments and future challenges. *IEEE Consumer Electronics Magazine*, 1(1), 2020.

**[12]** Shu Li, Jeong Geun Kim, Doo Hee Han, and Kye San Lee. A survey of energy-efficient communication protocols with qos guarantees in wireless multimedia sensor networks. *Sensors*, 19(1):1–29, 2019.

**[13]** Ahmed Mateen, Maida Sehar, Khizar Abbas, and Muhammad Azeem Akbar. Comparative analysis of wireless sensor networks with wireless multimedia sensor networks. In *International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, pages 80–83, 2017.

**[14]** Asim Rasheed, Saira Gillani, Sana Ajmal, and Amir Qayyum. Vehicular ad hoc network (vanet): A survey, challenges, and applications. In *Vehicular Ad-Hoc Networks for Smart Cities*, pages 39–51. Springer, 2017.

**[15]** Shaimaa M Mohamed, Haitham S Hamza, and Iman Aly Saroit. Coverage in mobile wireless sensor networks (m-wsn): A survey. *Computer Communications*, 110:133–150, 2017.

**[16]** Omar Sami Oubbati, Mohammed Atiquzzaman, Tariq Ahamed Ahanger, and Atef Ibrahim. Softwarization of uav networks: A survey of applications and future trends. *IEEE Access*, 8:98073–98125, 2020.

**[17]** Reza Shakeri, Mohammed Ali Al-Garadi, Ahmed Badawy, Amr Mohamed, Tamer Khattab, Abdulla Khalid Al-Ali, Khaled A Harras, and Mohsen Guizani. Design challenges of multi-uav systems in cyber-physical applications: A comprehensive survey and future directions. *IEEE Communications Surveys & Tutorials*, 21(4):3340–3385, 2019.

**[18]** Tarek Azzabi, Hassene Farhat, and Nabil Sahli. A survey on wireless sensor networks security issues and military specificities. In *International Conference on Advanced Systems and Electric Technologies (IC_ASET)*, pages 66–72, 2017.

**[19]** Stephen Russell and Tarek Abdelzaher. The internet of battlefield things: the next generation of command, control, communications and intelligence (c3i) decision-making. In *Conference Military Communications Conference (MILCOM)*, pages 737–742, 2018.

**[20]** Gerrit Hoogenboom. The georgia automated environmental monitoring network. Georgia Institute of Technology, 1993.

**[21]** Geoffrey Werner-Allen, Jeff Johnson, Mario Ruiz, Jonathan Lees, and Matt Welsh. Monitoring volcanic eruptions with a wireless sensor network. In *Proceeedings of the Second European Workshop on Wireless Sensor Networks*, pages 108–120, 2005.

**[22]** Kirk Martinez, Royan Ong, and Jane Hart. Glacsweb: a sensor network for hostile environments. In *First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks*, pages 81–87, 2004.

**[23]** Mrinai M Dhanvijay and Shailaja C Patil. Internet of things: A survey of enabling technologies in healthcare and its applications. *Computer Networks*, 153:113–131, 2019.

**[24]** Yazdan Ahmad Qadri, Ali Nauman, Yousaf Bin Zikria, Athanasios V Vasilakos, and Sung Won Kim. The future of healthcare internet of things: a survey of emerging technologies. *IEEE Communications Surveys & Tutorials*, 22(2):1121–1167, 2020.

**[25]** Jayoung Kim, Alan S Campbell, Berta Esteban-Fernández de Ávila, and Joseph Wang. Wearable biosensors for healthcare monitoring. *Nature biotechnology*, 37(4):389–406, 2019.

**[26]** Adam B Noel, Abderrazak Abdaoui, Tarek Elfouly, Mohamed Hossam Ahmed, Ahmed Badawy, and Mohamed S Shehata. Structural health monitoring using wireless sensor networks: A comprehensive survey. *IEEE Communications Surveys & Tutorials*, 19(3):1403–1423, 2017.

**[27]** Andrey I Vlasov, Pavel V Grigoriev, Aleksey I Krivoshein, Vadim A Shakhnov, Sergey S Filin, and Vladimir S Migalin. Smart management of technologies: Predictive maintenance of industrial equipment using wireless sensor networks. *Entrepreneurship and Sustainability Issues*, 6(2):489–502, 2018.

**[28]** Milan Erdelj, Nathalie Mitton, and Enrico Natalizio. Applications of industrial wireless sensor networks, 2013.

**[29]** Konstantin Mikhaylov, Jouni Tervonen, Joni Heikkilä, and Janne Känsäkoski. Wireless sensor networks in industrial environment: Real-life evaluation results. In *2nd Baltic Congress on Future Internet Communications*, pages 1–7, 2012.

**[30]** Pan Yi, Xiao Lizhi, and Zhang Yuanzhong. Remote real-time monitoring system for oil and gas well based on wireless sensor networks. In *2010 International Conference on Mechanic Automation and Control Engineering*, pages 2427–2429. IEEE, 2010.

**[31]** Food FAO. The future of food and agriculture–trends and challenges. *Annual Report 296*, 2017.

**[32]** Muhammad Shoaib Farooq, Shamyla Riaz, Adnan Abid, Kamran Abid, and Muhammad Azhar Naeem. A survey on the role of iot in agriculture for the implementation of smart farming. *IEEE Access*, 7:156237–156271, 2019.

**[33]** Muhammad Shoaib Farooq, Shamyla Riaz, Adnan Abid, Tariq Umer, and Yousaf Bin Zikria. Role of iot technology in agriculture: A systematic literature review. *Electronics*, 9(2):319–360, 2020.

**[34]** Hossein Ahmadzadeh, Ellips Masehian, and Masoud Asadpour. Modular robotic systems: Characteristics and applications. *Journal of Intelligent & Robotic Systems*, 81(3-4):317–357, 2016.

**[35]** Hossein Ahmadzadeh and Ellips Masehian. Modular robotic systems: Methods and algorithms for abstraction, planning, control, and synchronization. *Artificial Intelligence*, 223:27–64, 2015.

**[36]** United Nations. The ocean conference factsheet. *oceanconference.un.org*, pages 1–7, 2017.

**[37]** ARGO. Argo project. *http://www.argo.ucsd.edu/index.html*.

**[38]** Quazi Mamun. A qualitative comparison of different logical topologies for wireless sensor networks. *Sensors*, 12(11):14887–14913, 2012.

**[39]** Trupti Mayee Behera, Sushanta Kumar Mohapatra, Umesh Chandra Samal, Mohammad S Khan, Mahmoud Daneshmand, and Amir H Gandomi. Residual energy-based cluster-head selection in wsns for iot application. *IEEE Internet of Things Journal*, 6(3):5132–5139, 2019.

**[40]** Suparna Biswas, Jayita Saha, Tanumoy Nag, Chandreyee Chowdhury, and Sarmistha Neogy. A novel cluster head selection algorithm for energy-efficient routing in wireless sensor network. In *6th international conference on advanced computing (IACC)*, pages 588–593, 2016.

**[41]** R Raj Priyadarshini and N Sivakumar. Cluster head selection based on minimum connected dominating set and bi-partite inspired methodology for energy conservation in wsns. *Journal of King Saud University-Computer and Information Sciences*, 2018.

**[42]** Yousif Khalid Yousif, R Badlishah, N Yaakob, and A Amir. An energy efficient and load balancing clustering scheme for wireless sensor network (wsn) based on distributed approach. In *Journal of Physics: Conference Series*, volume 1019, pages 1–6. IOP Publishing, 2018.

**[43]** Sang H Kang. Energy optimization in cluster-based routing protocols for large-area wireless sensor networks. *Symmetry*, 11(1):1–18, 2019.

**[44]** Govind P Gupta. Improved cuckoo search-based clustering protocol for wireless sensor networks. *Procedia Computer Science*, 125:234–240, 2018.

**[45]** Amine Rais, Khalid Bouragba, and Mohammed Ouzzif. Routing and clustering of sensor nodes in the honeycomb architecture. *Journal of Computer Networks and Communications*, 2019.

**[46]** Hassan Harb, Abdallah Makhoul, Rami Tawil, and Ali Jaber. A suffix-based enhanced technique for data aggregation in periodic sensor networks. In *International wireless communications and mobile computing conference (IWCMC)*, pages 494–499, 2014.

**[47]** Jeril Kuriakose, V Amruth, and N Swathy Nandhini. A survey on localization of wireless sensor nodes. In *International Conference on Information Communication and Embedded Systems (ICICES2014)*, pages 1–6. IEEE, 2014.

**[48]** Chuan Zhu, Chunlin Zheng, Lei Shu, and Guangjie Han. A survey on coverage and connectivity issues in wireless sensor networks. *Journal of Network and Computer Applications*, 35(2):619–632, 2012.

**[49]** Sukhwinder Sharma, Rakesh Kumar Bansal, and Savina Bansal. Issues and challenges in wireless sensor networks. In *International Conference on Machine Intelligence and Research Advancement*, pages 58–62, 2013.

**[50]** Giuseppe Anastasi, Marco Conti, Mario Di Francesco, and Andrea Passarella. Energy conservation in wireless sensor networks: A survey. *Ad hoc networks*, 7(3):537–568, 2009.

**[51]** Mohamed-Lamine Messai. Classification of attacks in wireless sensor networks. *arXiv preprint arXiv:1406.4516*, 2014.

**[52]** Alexandru Lavric and Valentin Popa. Performance evaluation of lorawan communication scalability in large-scale wireless sensor networks. *Wireless Communications and Mobile Computing*, 2018.

**[53]** Louie Chan, Karina Gomez Chavez, Heiko Rudolph, and Akram Hourani. Hierarchical routing protocols for wireless sensor network: A compressive survey. *Wireless Networks*, 26(5):3291–3314, 2020.

**[54]** Aastha Maheshwari and Narottam Chand. A survey on wireless sensor networks coverage problems. In *Proceedings of 2nd International Conference on Communication, Computing and Networking*, pages 153–164. Springer, 2019.

**[55]** Uma Maheswari and Akila Cse. A survey on recent techniques for energy efficient routing in wsn. *International Journal of Sensors and Sensor Networks*, 6(1):8–15, 2018.

**[56]** Vishal Krishna Singh, Vivek Kumar Singh, and Manish Kumar. In-network data processing based on compressed sensing in wsn: A survey. *Wireless Personal Communications*, 96(2):2087–2124, 2017.

**[57]** Soroush Abbasian Dehkordi, Kamran Farajzadeh, Javad Rezazadeh, Reza Farahbakhsh, Kumbesan Sandrasegaran, and Masih Abbasian Dehkordi. A survey on data aggregation techniques in iot sensor networks. *Wireless Networks*, 26(2):1243–1263, 2020.

**[58]** Jing Zhang, Zhiwei Lin, Pei-Wei Tsai, and Li Xu. Entropy-driven data aggregation method for energy-efficient wireless sensor networks. *Information Fusion*, 56:103–113, 2020.

**[59]** Ben Othman Soufiene, Abdullah Ali Bahattab, Abdelbasset Trad, and Habib Youssef. Lightweight and confidential data aggregation in healthcare wireless sensor networks. *Transactions on Emerging Telecommunications Technologies*, 27(4):576–588, 2016.

**[60]** Dan Popescu, Florin Stoican, Grigore Stamatescu, Loretta Ichim, and Cristian Dragana. Advanced uav–wsn system for intelligent monitoring in precision agriculture. *Sensors*, 20(3):1–25, 2020.

**[61]** Jacques M Bahi, Abdallah Makhoul, and Maguy Medlej. A two tiers data aggregation scheme for periodic sensor networks. *Ad Hoc & Sensor Wireless Networks*, 21(1-2):77–100, 2014.

**[62]** Deqing Wang, Ru Xu, Xiaoyi Hu, and Wei Su. Energy-efficient distributed compressed sensing data aggregation for cluster-based underwater acoustic sensor networks. *International Journal of Distributed Sensor Networks*, 12(3):1–14, 2016.

**[63]** Deqing Wang, Ru Xu, and Xiaoyi Hu. Energy-efficient data aggregation scheme for underwater acoustic sensor networks. In *Proceedings of the 10th International Conference on Underwater Networks & Systems*, pages 1–2, 2015.

**[64]** Yao Lu, Ioan Sorin Comsa, Pierre Kuonen, and Beat Hirsbrunner. Dynamic data aggregation protocol based on multiple objective tree in wireless sensor networks. In *Tenth international conference on intelligent sensors, sensor networks and information processing (ISSNIP)*, pages 1–7, 2015.

**[65]** Yung-Kuei Chiang, Neng-Chung Wang, and Chih-Hung Hsieh. A cycle-based data aggregation scheme for grid-based wireless sensor networks. *Sensors*, 14(5):8447–8464, 2014.

**[66]** De-gan Zhang, Ting Zhang, Jie Zhang, Yue Dong, and Xiao-dan Zhang. A kind of effective data aggregating method based on compressive sensing for wireless sensor network. *EURASIP Journal on Wireless Communications and Networking*, 2018(1):1–15, 2018.

**[67]** Peng Zeng, Bofeng Pan, Kim-Kwang Raymond Choo, and Hong Liu. Mmda: Multidimensional and multidirectional data aggregation for edge computing-enhanced iot. *Journal of Systems Architecture*, 106:1–9, 2020.

**[68]** S Pushpalatha and KS Shivaprakasha. Energy-efficient communication using data aggregation and data compression techniques in wireless sensor networks: A survey. In *Advances in communication, signal processing, VLSI, and embedded systems*, pages 161–179. Springer, 2020.

**[69]** Ikjune Yoon and Dong Kun Noh. Energy-aware control of data compression and sensing rate for wireless rechargeable sensor networks. *Sensors*, 18(8):1–18, 2018.

**[70]** Yao Liang and Yimei Li. An efficient and robust data compression algorithm in wireless sensor networks. *IEEE Communications Letters*, 18(3):439–442, 2014.

**[71]** Qinbao Xu, Rizwan Akhtar, Xing Zhang, and Changda Wang. Cluster-based arithmetic coding for data provenance compression in wireless sensor networks. *Wireless Communications and Mobile Computing*, 2018:1–15, 2018.

**[72]** Hongzhi Lin, Wei Wei, Ping Zhao, Xiaoqiang Ma, Rui Zhang, Wenping Liu, Tianping Deng, and Kai Peng. Energy-efficient compressed data aggregation in underwater acoustic sensor networks. *Wireless networks*, 22(6):1985–1997, 2016.

**[73]** Yihang Yin, Fengzheng Liu, Xiang Zhou, and Quanzhong Li. An efficient data compression model based on spatial clustering and principal component analysis in wireless sensor networks. *Sensors*, 15(8):19443–19465, 2015.

**[74]** Matteo Gaeta, Vincenzo Loia, and Stefania Tomasiello. Multisignal 1-d compression by f-transform for wireless sensor networks applications. *Applied Soft Computing*, 30:329–340, 2015.

**[75]** Matteo Gaeta, Vincenzo Loia, and Stefania Tomasiello. Cubic b–spline fuzzy transforms for an efficient and secure compression in wireless sensor networks. *Information Sciences*, 339–350:19–30, 2016.

**[76]** Joyce Chiang and Rabab K Ward. Energy-efficient data reduction techniques for wireless seizure detection systems. *Sensors*, 14(2):2036–2051, 2014.

**[77]** Gabriel Martins Dias, Boris Bellalta, and Simon Oechsner. A survey about prediction-based data reduction in wireless sensor networks. *ACM Computing Surveys (CSUR)*, 49(3):1–35, 2016.

**[78]** Hidaya Liazid, Mohamed Lehsaini, and Abdelkrim Liazid. An improved adaptive dual prediction scheme for reducing data transmission in wireless sensor networks. *Wireless Networks*, 25(6):3545–3555, 2019.

**[79]** Guangjie Han, Songjie Shen, Hao Wang, Jinfang Jiang, and Mohsen Guizani. Prediction-based delay optimization data collection algorithm for underwater acoustic sensor networks. *IEEE Transactions on Vehicular Technology*, 68(7):6926–6936, 2019.

**[80]** Uélison Jean L dos Santos, Gustavo Pessin, Cristiano André da Costa, and Rodrigo da Rosa Righi. Agriprediction: A proactive internet of things model to anticipate problems and improve production in agricultural crops. *Computers and electronics in agriculture*, 161:202–213, 2019.

**[81]** Xiaobin Xu and Guangwei Zhang. A hybrid model for data prediction in real-world wireless sensor networks. *IEEE Communications Letters*, 2017.

**[82]** Samer Samarah. Vector-based data prediction model for wireless sensor networks. *International Journal of High Performance Computing and Networking*, 9(4):310–315, 2016.

**[83]** Xiaobin Xu. Data approximation for time series data in wireless sensor networks. *International Journal of Data Warehousing and Mining (IJDWM)*, 12(3):1–13, 2016.

**[84]** Adrien Russo, François Verdier, and Benoît Miramond. Energy saving in a wireless sensor network by data prediction by using self-organized maps. *Procedia computer science*, 130:1090–1095, 2018.

**[85]** Md Monirul Islam, Zabir Al Nazi, ABM Aowlad Hossain, Md Masud Rana, et al. Data prediction in distributed sensor networks using adam bashforth moulton method. *Journal of Sensor Technology*, 8(2):48–57, 2018.

**[86]** Md Monirul Islam, Zabir Al Nazi, Md Masud Rana, and ABM Aowlad Hossain. Information prediction in sensor networks using milne-simpson's scheme. In *2017 4th International Conference on Advances in Electrical Engineering (ICAEE)*, pages 494–498. IEEE, 2017.

**[87]** Suat Ozdemir, Miao Peng, and Yang Xiao. Prda: polynomial regression-based privacy-preserving data aggregation for wireless sensor networks. *Wireless communications and mobile computing*, 15(4):615–628, 2015.

**[88]** Gopal Krishna, Sunil Kumar Singh, Jyoti Prakash Singh, and Prabhat Kumar. Energy conservation through data prediction in wireless sensor networks. In *Proceedings of 3rd International Conference on Internet of Things and Connected Technologies (ICIoTCT)*, pages 26–27, 2018.

**[89]** Usman Raza, Alessandro Camerra, Amy L Murphy, Themis Palpanas, and Gian Pietro Picco. Practical data prediction for real-world wireless sensor networks. *IEEE Transactions on Knowledge and Data Engineering*, 27(8):2231–2244, 2015.

**[90]** Jyotirmoy Karjee and Martin Kleinsteuber. Data estimation with predictive switching mechanism in wireless sensor networks. *International Journal of Sensor Networks*, 25(3):184–197, 2017.

**[91]** Sivadi Balakrishna and M Thirumaran. Semantics and clustering techniques for iot sensor data analysis: A comprehensive survey. *Principles of Internet of Things (IoT) Ecosystem: Insight Paradigm*, pages 103–125, 2020.

**[92]** Siguang Chen, Shujun Zhang, Xiaoyao Zheng, and Xiukai Ruan. Layered adaptive compression design for efficient data collection in industrial wireless sensor networks. *Journal of Network and Computer Applications*, 129:37–45, 2019.

**[93]** G Pius Agbulu, G Joselin Retna Kumar, and A Vimala Juliet. A lifetime-enhancing cooperative data gathering and relaying algorithm for cluster-based wireless sensor networks. *International Journal of Distributed Sensor Networks*, 16(2):1–12, 2020.

**[94]** Zhiping Wan, Shaojiang Liu, Weichuan Ni, and Zhiming Xu. An energy-efficient multi-level adaptive clustering routing algorithm for underwater wireless sensor networks. *Cluster Computing*, 22(6):14651–14660, 2019.

**[95]** Vani Krishnaswamy and Sunilkumar S Manvi. Fuzzy and pso based clustering scheme in underwater acoustic sensor networks using energy and distance parameters. *Wireless Personal Communications*, 108(3):1529–1546, 2019.

**[96]** Elham Moridi, Majid Haghparast, Mehdi Hosseinzadeh, and Somaye Jafarali Jassbi. Novel fault-tolerant clustering-based multipath algorithm (ftcm) for wireless sensor networks. *Telecommunication Systems*, 74(4):411–424, 2020.

**[97]** V Pandiyaraju, R Logambigai, Sannasi Ganapathy, and Arputharaj Kannan. An energy efficient routing algorithm for wsns using intelligent fuzzy rules in precision agriculture. *Wireless Personal Communications*, pages 1–17, 2020.

**[98]** Kashif Naseer Qureshi, Muhammad Umair Bashir, Jaime Lloret, and Antonio Leon. Optimized cluster-based dynamic energy-aware routing protocol for wireless sensor networks in agriculture precision. *Journal of sensors*, 2020, 2020.

**[99]** Khoa Thi-Minh Tran and Seung-Hyun Oh. Uwsns: A round-based clustering scheme for data redundancy resolve. *International Journal of Distributed Sensor Networks*, 10(4):1–6, 2014.

**[100]** Xiaoyan Kui, Jianxin Wang, Shigeng Zhang, and JLANNONG CAO. Energy balanced clustering data collection based on dominating set in wireless sensor networks. *Adhoc & Sensor Wireless Networks*, 24, 2015.

**[101]** Pinghui Zou and Yun Liu. A data-aggregation scheme for wsn based on optimal weight allocation. *Journal of Networks*, 9(1):100, 2014.

**[102]** Faycal Bouhafs, Madjid Merabti, and Hala M Mokhtar. A semantic clustering routing protocol for wireless sensor networks. In *3rd Consumer Communications and Networking Conference*, pages 351–355, 2006.

**[103]** Tao Du, Zhe Qu, Qingbei Guo, and Shouning Qu. A high efficient and real time data aggregation scheme for wsns. *International journal of distributed sensor networks*, 11(6):261381, 2015.

**[104]** Atslands R Rocha, Luci Pirmez, Flávia C Delicato, Érico Lemos, Igor Santos, Danielo G Gomes, and José Neuman de Souza. Wsns clustering based on semantic neighborhood relationships. *Computer Networks*, 56(5):1627–1645, 2012.

**[105]** Hassan Harb, Abdallah Makhoul, and Raphaël Couturier. An enhanced k-means and anova-based clustering approach for similarity aggregation in underwater wireless sensor networks. *IEEE Sensors Journal*, 15(10):5483–5493, 2015.

**[106]** Khoa Thi-Minh Tran, Seung-Hyun Oh, and Jeong-Yong Byun. Well-suited similarity functions for data aggregation in cluster-based underwater wireless sensor networks. *International Journal of Distributed Sensor Networks*, 9(8):1–7, 2013.

**[107]** Mohammad Fajar, Junishia Litan, Abdul Munir, Agus Halid, et al. Energy efficiency using data filtering approach on agricultural wireless sensor network. *International Journal of Computer Engineering and Information Technology*, 9(9):192, 2017.

**[108]** Kojo Sarfo Gyamfi, James Brusey, Andrew Hunt, and Elena Gaura. Linear dimensionality reduction for classification via a sequential bayes error minimisation with an application to flow meter diagnostics. *Expert Systems with Applications*, 91:252–262, 2018.

**[109]** Hassan Harb, Abdallah Makhoul, Raphaël Couturier, and Maguy Medlej. Atp: An aggregation and transmission protocol for conserving energy in periodic sensor networks. In *24th International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises*, pages 134–139, 2015.

**[110]** Carol Habib, Abdallah Makhoul, Rony Darazi, and Christian Salim. Self-adaptive data collection and fusion for health monitoring based on body sensor networks. *IEEE Transactions on Industrial Informatics*, 12(6):2342–2352, 2016.

**[111]** Abdallah Makhoul, David Laiymani, Hassan Harb, and Jacques M Bahi. An adaptive scheme for data collection and aggregation in periodic sensor networks. *International journal of sensor networks*, 18(1-2):62–74, 2015.

**[112]** Hassan Harb, Abdallah Makhoul, David Laiymani, Ali Jaber, and Rami Tawil. K-means based clustering approach for data aggregation in periodic sensor networks. In *10th international conference on wireless and mobile computing, networking and communications (WiMob)*, pages 434–441, 2014.

**[113]** Shahinaz M Al-Tabbakh. Novel technique for data aggregation in wireless sensor networks. In *International conference on internet of things, embedded systems and communications (IINTEC)*, pages 1–8, 2017.

**[114]** Bugra Gedik, Ling Liu, and S Yu Philip. Asap: An adaptive sampling approach to data collection in sensor networks. *IEEE Transactions on Parallel and distributed systems*, 18(12):1766–1783, 2007.

**[115]** Jinseok Yang, Sameer Tilak, and Tajana Simunic Rosing. An interactive context-aware power management technique for optimizing sensor network lifetime. In *SENSORNETS*, pages 69–76, 2016.

**[116]** David Laiymani and Abdallah Makhoul. Adaptive data collection approach for periodic sensor networks. In *2013 9th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 1448–1453. IEEE, 2013.

**[117]** Mehmet Başaran, Stephan Schlupkothen, and Gerd Ascheid. Adaptive sampling techniques for autonomous agents in wireless sensor networks. In *IEEE 30th annual international symposium on personal, indoor and mobile radio communications (PIMRC)*, pages 1–6, 2019.

**[118]** Ankur Jain and Edward Y Chang. Adaptive sampling for sensor networks. In *Proceedings of the 1st international workshop on Data management for sensor networks*, pages 10–16, 2004.

**[119]** Mehmet C Vuran and Ian F Akyildiz. Spatial correlation-based collaborative medium access control in wireless sensor networks. *IEEE/ACM Transactions On Networking*, 14(2):316–329, 2006.

**[120]** Hassan Harb and Abdallah Makhoul. Energy-efficient sensor data collection approach for industrial process monitoring. *IEEE Transactions on Industrial Informatics*, 14(2):661–672, 2017.

**[121]** Maryam Vahabi, M Rasid, RSAR Abdullah, and M Ghazvini. Adaptive data collection algorithm for wireless sensor networks. *International Journal of Computer Science and Network Security*, 8(6):1–13, 2008.

**[122]** Guangjie Han, Zhengkai Tang, Yu He, Jinfang Jiang, and James Adu Ansere. District partition-based data collection algorithm with event dynamic competition in underwater acoustic sensor networks. *IEEE Transactions on Industrial Informatics*, 15(10):5755–5764, 2019.

**[123]** Fekher Khelifi. Monitoring system based in wireless sensor network for precision agriculture. In *Internet of Things (IoT)*, pages 461–472. Springer, 2020.

**[124]** Alireza Masoum, Nirvana Meratnia, and Paul JM Havinga. An energy-efficient adaptive sampling scheme for wireless sensor networks. In *Eighth International Conference on Intelligent Sensors, Sensor Networks and Information Processing*, pages 231–236, 2013.

**[125]** Alex L Wood, Geoff V Merrett, Steve R Gunn, Bashir M Al-Hashimi, Nigel R Shadbolt, and Wendy Hall. Adaptive sampling in context-aware systems: a machine learning approach. 2012.

**[126]** Revathi Sundarasekar, P Mohamed Shakeel, S Baskar, Seifedine Kadry, George Mastorakis, Constandinos X Mavromoustakis, R Dinesh Jackson Samuel, and Vivekananda Gn. Adaptive energy aware quality of service for reliable data transfer in under water acoustic sensor networks. *IEEE Access*, 7:80093–80103, 2019.

**[127]** Ajit R Pagar and DC Mehetre. A survey on energy efficient sleep scheduling in wireless sensor network. *International Journal of Advanced Research in Computer Science and Software Engineering*, 5(1):557–562, 2015.

**[128]** K Karuppasamy and V Gunaraj. Optimizing sensing quality with coverage and lifetime in wireless sensor networks. *International Journal of Engineering Research and Technology*, 2(2):1–7, 2013.

**[129]** Mou Wu, Liansheng Tan, and Naixue Xiong. A structure fidelity approach for big data collection in wireless sensor networks. *Sensors*, 15(1):248–273, 2015.

**[130]** Sunil Dhimal and Kalpana Sharma. Energy conservation in wireless sensor networks by exploiting inter-node data similarity metrics. *International Journal of Energy, Information and Communications*, 6(2):23–32, 2015.

**[131]** Rongrong Zhang and Jihong Yu. Energy-efficient and reliable sleep scheduling algorithms in wbsns. In *Energy-Efficient Algorithms and Protocols for Wireless Body Sensor Networks*, pages 101–121. Springer, 2020.

**[132]** Md Khurram Monir Rabby, Mohammad Shah Alam, and MST Shamim Ara Shawkat. A priority based energy harvesting scheme for charging embedded sensor nodes in wireless body area networks. *PloS one*, 14(4):1–22, 2019.

**[133]** Leandro A Villas, Azzedine Boukerche, Daniel L Guidoni, Horacio ABF De Oliveira, Regina Borges De Araujo, and Antonio AF Loureiro. An energy-aware spatio-temporal correlation mechanism to perform efficient data collection in wireless sensor networks. *Computer Communications*, 36(9):1054–1066, 2013.

**[134]** Fernando R Almeida, Angelo Brayner, Joel JPC Rodrigues, and Jose E Bessa Maia. Improving multidimensional wireless sensor network lifetime using pearson correlation and fractal clustering. *Sensors*, 17(6):1317, 2017.

**[135]** Andrei BB Torres, Atslands R da Rocha, Ticiana L Coelho da Silva, José N de Souza, and Rubens S Gondim. Multilevel data fusion for the internet of things in smart agriculture. *Computers and Electronics in Agriculture*, 171:1–16, 2020.

**[136]** P Vignesh Raja and E Sivasankar. Modern framework for distributed healthcare data analytics based on hadoop. In *Information and Communication Technology-EurAsia Conference*, pages 348–355. Springer, 2014.

**[137]** Mahammad Shareef Mekala and P Viswanathan. (t, n): Sensor stipulation with tham index for smart agriculture decision-making iot system. *Wireless Personal Communications*, 111(3):1909–1940, 2020.

**[138]** Tian Wang, Md Zakirul Alam Bhuiyan, Guojun Wang, Md Arafatur Rahman, Jie Wu, and Jiannong Cao. Big data reduction for a smart city's critical infrastructural health monitoring. *IEEE Communications Magazine*, 56(3):128–133, 2018.

**[139]** Abdalla Alameen and Ashu Gupta. Clustering and classification based real time analysis of health monitoring and risk assessment in wireless body sensor networks. *Bio-Algorithms and Med-Systems*, 15(4), 2019.

**[140]** Maroun Koussaifi, Carol Habib, and Abdallah Makhoul. Real-time stress evaluation using wireless body sensor networks. In *Wireless Days (WD)*, pages 37–39, 2018.

**[141]** Sen Qiu, Long Liu, Zhelong Wang, Shengming Li, Hongyu Zhao, Jiaxin Wang, Jinxiao Li, and Kai Tang. Body sensor network-based gait quality assessment for clinical decision-support via multi-sensor fusion. *IEEE Access*, 7:59884–59894, 2019.

**[142]** Kaushik Sekaran, Maytham N Meqdad, Pardeep Kumar, Soundar Rajan, and Seifedine Kadry. Smart agriculture management system using internet of things. *Telkomnika*, 18(3):1275–1284, 2020.

**[143]** Ying Liu, Lin Zhang, Yuan Yang, Longfei Zhou, Lei Ren, Fei Wang, Rong Liu, Zhibo Pang, and M Jamal Deen. A novel cloud-based framework for the elderly healthcare services using digital twin. *IEEE Access*, 7:49088–49101, 2019.

**[144]** Thomas W Rauber, Eduardo Mendel do Nascimento, Estefhan D Wandekokem, Flávio M Varejão, and A Herout. Pattern recognition based fault diagnosis in industrial processes: Review and application. *Pattern Recognition Recent Advances*, pages 483–508, 2010.

**[145]** Madden Madden. Intel berkeley research lab. *http://db.csail.mit.edu/labdata/labdata.html*, 2004.

**[146]** Advanticsys. Online data. *http://www.advanticsys.com/wiki/index.php? title=sg1000*.

**[147]** David Gay, Philip Levis, David Culler, and Eric Brewer. nesc language manual: https://github.com/tinyos/nesc/blob/master/doc/ref.pdf?raw=true. 2009.

**[148]** Charu C Aggarwal. *Data mining: the textbook*. Springer, 2015.

**[149]** Wendi Beth Heinzelman. *Application-specific protocol architectures for wireless networks*. PhD thesis, Massachusetts Institute of Technology, June 2000.

**[150]** Douglas H Jones. Book review: Statistical methods. *Journal of Educational Statistics*, 19(3):304–307, 1994.

**[151]** Abdallah Makhoul, Hassan Harb, and David Laiymani. Residual energy-based adaptive data collection approach for periodic sensor networks. *Ad Hoc Networks*, 35:149–160, 2015.

**[152]** James MacQueen et al. Some methods for classification and analysis of multivariate observations. In *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, volume 1, pages 281–297. Oakland, CA, USA, 1967.

**[153]** Paulene Govender and Venkataraman Sivakumar. Application of k-means and hierarchical clustering techniques for analysis of air pollution: A review (1980–2019). *Atmospheric Pollution Research*, 11(1):40–56, 2020.

**[154]** K Lavanya, Rani Kashyap, S Anjana, and Sumaiya Thasneen. An enhanced k-means msoinn based clustering over neo4j with an application to weather analysis. In *International Conference on Intelligent Computing and Smart Communication*, pages 451–461, 2020.

**[155]** Guiqing Zhang, Yong Li, and Xiaoping Deng. K-means clustering-based electrical equipment identification for smart building application. *Information*, 11(1):1–27, 2020.

**[156]** statista. Internet of things (iot) and non-iot active device connections worldwide from 2010 to 2025. *https://www.statista.com/statistics/1101442/iot-number-of-connected-devices-worldwide/*, 2021.

**[157]** Debashis De, Amartya Mukherjee, Santosh Kumar Das, and Nilanjan Dey. Wireless sensor network: applications, challenges, and algorithms. In *Nature Inspired Computing for Wireless Sensor Networks*, pages 1–18. Springer, 2020.

**[158]** Hassan Harb, Ali K Idrees, Ali Jaber, Abdallah Makhoul, Oussama Zahwe, and Mohamad Abou Taam. Wireless sensor networks: A big data source in internet of things. *International Journal of Sensors Wireless Communications and Control*, 7(2):93–109, 2017.

**[159]** Abdul Majeed and Ibtisam Rauf. Graph theory: A comprehensive survey about graph theory applications in computer science and social networks. *Inventions*, 5(1):10, 2020.

**[160]** Debao Chen, Feng Zou, Renquan Lu, and Peng Wang. Learning backtracking search optimisation algorithm and its application. *Information Sciences*, 376:71–94, 2017.

**[161]** Bryar A Hassan and Tarik A Rashid. Operational framework for recent advances in backtracking search optimisation algorithm: A systematic review and performance evaluation. *Applied Mathematics and Computation*, 370:124919, 2020.

**Titre :** Vers des Stratégies Efficaces de Collecte de Données et de Prise de Décision pour les Réseaux de Capteurs à Ressources Limitées

**Mots clés :** Réseaux des Capteurs, Efficacité Energétique, Prise de Décision, Réduction des Données, Corrélation Spatio-Temporelle, Stratégies de Planification.

**Résumé :** Bien que les avantages potentiels de la technologie de capteurs soient réels et importants, deux défis majeurs restent à relever pour réaliser pleinement ce potentiel : les ressources limitées de capteurs, en particulier la puissance de la batterie, et la prise de décision dans les applications de temps réel. Dans cette thèse, nous proposons plusieurs mécanismes de collecte et d'analyse de données qui permettent de surmonter les ressources limitées de capteurs et les défis de collecte de données volumineux imposés par les réseaux de capteurs, en se basant sur l'architecture clustering de réseaux. Principalement, les mécanismes proposés fonctionnent à trois niveaux de réseau (par exemple, capteur, CH et puits), et ils visent à réduire la quantité de données disséminées dans le réseau tout en préservant l'intégrité des informations au niveau du puits. Au niveau du capteur, nous proposons des méthodes de prédiction, d'agrégation et de compression de données basées respectivement sur des algorithmes de Newton Forward Difference, de divide-and-conquer et d'élimination de similarité dans le but de réduire les données brutes collectées par chaque capteur. Au niveau de CH, nous proposons de nouvelles techniques de clustering, de fusion, d'agrégation intermédiaire et d'ordonnancement qui visent à rechercher la corrélation entre les nœuds voisins puis à éliminer les redondances de données existantes avant d'envoyer les données vers le puits. Au niveau du puits, nous introduisons des modèles de prise de décision efficaces basés sur un tableau de score qui permet aux utilisateurs finaux d'analyser les données et de prendre une décision convenable. Nous avons évalué les performances de nos mécanismes en se basant sur de simulations et d'expérimentations. Les résultats obtenus ont montré l'efficacité de nos mécanismes en terme de la consommation d'énergie, de la précision des données et de la zone de couverture tout en améliorant les performances des réseaux de capteurs.

**Title:** Toward Efficient Data Collection and Decision-Making Strategies for Resource-Constrained Sensor Networks

**Keywords:** Sensing-based Networks, Energy-Efficiency, Decision-Making, Data Reduction, Spatio-Temporal Correlation, Scheduling Strategies.

**Abstract:** While the potential benefits of sensing-based technology is real and significant, two major challenges remain in front of fully realizing this potential: resource-constrained sensors, especially the battery power, and decision making in real-time applications. In this thesis, we propose several data collection and analysis mechanisms that allow overcoming the limited sensor resources and the big data collection challenges imposed by sensing-based networks, under the clustering-based network architecture. Mainly, the proposed mechanisms work on three network levels (e.g. sensor, CH and sink), and they aim to reduce the amount of data routed in the network while preserving the information integrity at the sink. At the sensor level, we propose data prediction, aggregation and compression methods based respectively on Newton forward difference, divide-and-conquer and elimination similarity algorithms with the aim to reduce the raw data collected by each sensor. At the CH level, we propose new data clustering, fusion, in-network aggregation and scheduling techniques that aim to search the correlation among neighbouring nodes then to eliminate the existing data redundancies before sending the data toward the sink. At the sink level, we introduce efficient decision-making models based on customizable user-defined tables that allow end users to analyse the data and make an early decision. We analysed the performance of our mechanisms based on a set of simulation and experimentations. The obtained results have shown the efficiency of our mechanisms according to energy consumption, data accuracy, and coverage area while improving the performance of sensing-based networks.